

Ingénierie des systèmes complexes

Méthodologie de design



Denis Laurendeau
Professeur
Génie électrique et génie informatique
Université Laval

Note: ces notes s'inspirent du cours:

« Systems Engineering Training »,
Joe Jenney and Scott Armstrong
ITT Industries, Fort Wayne, IN, USA

Table des matières

- Historique des processus d'ingénierie des systèmes
- Approche d'ingénierie simultanée (« concurrent design »)
 - Les étapes en bref
 - Etape 1: Focalisation de l'équipe de design
 - Etape 2: Planification du projet
 - Etape 3: Développement du baseline du système
 - Etape 4: Développement des sous-systèmes
 - Etape 5: Définition des performances du système
 - Etape 6: Vérification des performances du système
 - Etape 7: Intégration du système

Ingénierie des systèmes complexes

- Historique
 - Avant 1940
 - 1940-1980
 - 1980-20??

Historique

- Avant 1940
 - Projets simples
 - Formation de l'équipe:
 - Un ingénieur chef de projet
 - Une équipe de développeurs
 - Avantages:
 - Communication facile à l'intérieur de l'équipe
 - Inconvénients:
 - Approche limitée aux systèmes simples

Historique des approches de design de systèmes

- 1940-1980
 - Projets plus complexes
 - Formation de l'équipe:
 - Ingénieurs de systèmes et équipe de développement formée de *spécialistes*
 - Approche séquentielle de design avec *revues de design* périodiques.
 - Avantages de cette approche:
 - Possibilité de concevoir des projets plus gros grâce à l'ajout de spécialistes du domaine
 - Inconvénients de l'approche:
 - Les longues périodes de temps entre les revues de design ne permettent pas de découvrir les faiblesses du design assez tôt

Historique (suite)

- Le processus séquentiel tente d'estimer simultanément le coût, l'échéancier et la qualité du système. Seulement deux de ces quantités sont indépendantes
- Le produit arrive à l'étape de la fabrication en série (manufacturing) avant que la méthodologie de fabrication ne soit complètement terminée

Historique (suite)

- 1980-20??
 - Nouvelles méthodes de *d'ingénierie simultanée* afin de contourner les difficultés propres à l'approche séquentielle
 - Approche:
 - Développement *simultané* des sous-systèmes
 - Implication hâtive des différents *intervenants* dans la conception et la fabrication du système:
 - » Manufacturing
 - » Contrôle de qualité
 - » Service des achats
 - » Fournisseurs
 - » Etc.
 - Objectif:
 - *Prévenir* les problèmes plutôt que les solutionner
 - Concevoir le *bon système* dès la *première itération* (do it right the first time!)

Approche d'ingénierie simultanée des systèmes (Concurrent Engineering)

- Les étapes du processus moderne de conception des systèmes complexes
 - *Focalisation* de l'équipe
 - *Planification* du projet
 - Développement du *baseline*
 - Développement des *sous-systèmes* (allocated baseline)
 - *Définition* des *performances*
 - *Vérification* des *performances*
 - *Intégration* des sous-systèmes
 - Support à la *production*

L'approche d'ingénierie simultanée de systèmes complexes



Etape 1

Focalisation de l'équipe de design

Etape 1: focalisation de l'équipe de design...

- Activités de cette étape:
 - (*)Etablir les *rôles* et les *responsabilités* de chaque membre de l'équipe
 - (*)Concevoir un *diagramme de contexte* décrivant l'essence du système à réaliser
 - (*)Utiliser le concept de *DPF* (Description des Propriétés Fonctionnelles) pour établir clairement les exigences du *client* et la proposition de l'équipe de développement (le *fournisseur*) pour répondre à ces exigences
 - Concevoir des *diagrammes de Kano* pour maximiser les chances de succès et anticiper les développements futurs

(*) Les items marqués d'une étoile sont à considérer dans le cadre du cours de Design III)

Etape 1: focalisation de l'équipe de design (suite)

- Diagramme de contexte:
 - *Isole* le système du reste de l'univers
 - *Nomme* le système
 - *Cache* les *parties internes* au système
 - Met en évidence les *entrées* fournies au système et les *actions effectuées par celui-ci* en forçant leur identification et leur définition
 - Rend la *fonctionnalité principale* du système évidente
 - Permet de *vérifier* avec le client l'exactitude de la compréhension du problème par l'équipe de design
 - Plusieurs *itérations* sont nécessaires pour arriver à un diagramme complet. Des révisions sont nécessaires (et bénéfiques).
 - Exemple de [diagramme de contexte](#) pour la conception d'un système d'alarme

Etape 1: ... Concept de DPF

- *Une DPF est un outil utile pour formuler un problème*
- *La DPF focalise l'équipe sur les exigences et les attentes du client*
- *La DPF identifie les fonctionnalités (et sous-fonctionnalités) que doit offrir le système pour satisfaire les exigences et les paramètres critiques (PC) associés à chaque fonctionnalité*
- *Les fonctionnalités sont décrites par des verbes d'action*
- *La DPF doit être utilisée tout au long du cycle de développement du système*
- *La DPF ne tient pas immédiatement compte des risques associés à chaque fonctionnalité. Voir notions de risque et de registre de risques p. 43.*

Etape 1 ... Comment concevoir une DPF

- *Remue-méninges sur **qui** est le client et les autres parties impliquées dans le projet*
- *Remue-méninges sur les **exigences** du client*
- *Classer les **exigences** en ordre de **priorité***
- *Remue-méninges sur les **forces** de la compagnie et sur les **fonctionnalités** requises pour le produit (voir p. 30 pour une définition précise de « fonctionnalité »)*
- *Organiser ces **fonctionnalités** en catégories (et leur associer les paramètres critiques)*
- *Construire une **matrice de relations** entre les exigences du client et les fonctionnalités*
- *Tenter d'estimer les **valeurs souhaitées de performance** pour les fonctionnalités (dans le cadre du cours, cette étape n'est pas couverte)*

Etape 1 ... Concept de DPF (suite)

Structure d'une DPF

Matrice de couplage					P e r c e p t i o n
Comment? Quoi?	Fonctionnalités nécessaires pour satisfaire les exigences				
	Catégorie 1		Catégorie 2		
	Fonct. 1 Paramètres critiques	Fonct. 2 Paramètres critiques	Fonct. 1 Paramètres critiques	Fonct. 2 Paramètres critiques	
Exigence 1	2		5		
Exigence 2					
Exigences du client	Matrice de relations				
Combien?					
Estimation des avantages compétitifs					

Nombre entre 0 et 5 estimant la dépendance entre une exigence et une fonctionnalité

Voir matrice de couplage page suivante

Exemple d'une DPF pour le Système d'alarme

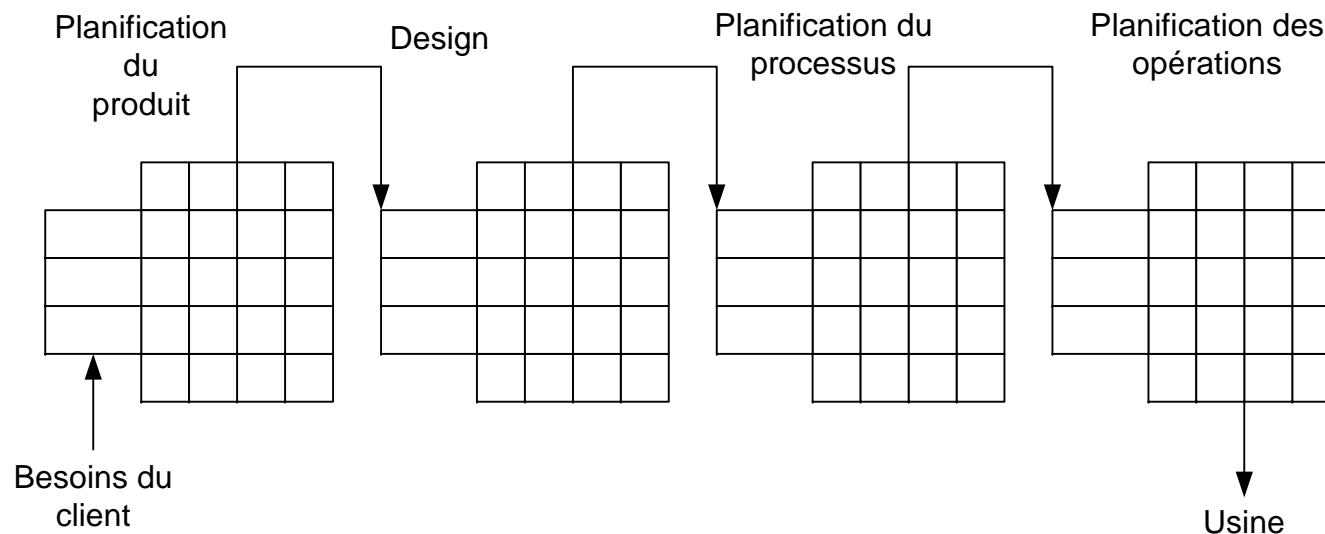
Etape 1 ... Concept de DPF (suite)

Matrice de couplage

		<i>Cat. 1</i>		<i>Cat 2</i>	
		Fonct. 1	Fonct. 2	Fonct. 1	Fonct. 2
<i>Cat. 1</i>	Fonct. 1				
	Fonct. 2				
	Fonct. 1				
	Fonct. 2				

Etape 1 ... Concept de DPF (suite)

Dans les systèmes complexes pour lesquels le projet doit tenir compte de la fabrication, les fonctionnalités d'une étape deviennent les exigences de l'étape suivante



Etape 1 ... Construction de diagrammes de Kano

- Les **diagrammes de Kano** permettent d'identifier le **design gagnant** auprès du client.
- Ils illustrent graphiquement le **niveau de satisfaction** du client en fonction de la **présence** ou de l'**absence** de caractéristiques sur le produit
- Ce niveau de satisfaction se trace pour trois cas:
 - Caractéristiques **obligatoires**
 - Caractéristiques **obligatoires améliorées** et caractéristiques **optionnelles mineures**
 - Caractéristiques **optionnelles majeures**



Etape 2

Planification du projet

Etape 2: Planification du projet

- *Principaux buts de cette étape*
 - **Identifier** et **documenter** les **tâches techniques critiques** qui doivent être implantées pour mener à un projet fonctionnel
 - **Identifier** et **documenter** les autres facteurs critiques au fonctionnement du projet
- *Approche pour la planification du projet*
 - Développement d'un **plan de gestion intégré (PGI!)**
 - (*) Développement d'un **diagramme de flux des activités techniques** (PERT flow chart et [Gantt Chart](#))
 - Développement d'un **plan de modélisation et de simulation (PMS)**

Etape 2 ... Plan de Gestion Intégré (PGI)

- *Le PGI est formé des éléments suivants:*
 - *Une description de haut niveau du projet*
 - *(*)Liste des biens livrables critiques (milestones)*
 - *(*)Description des biens livrables et les standards de qualité associés*
 - *Liste des objectifs à long terme (par rapport à la gamme des produits déjà supportés par l'entreprise)*
 - *Diagramme PERT au niveau système*
 - *Un arbre des plans de haut niveau*
 - *Hardware/software*
 - *Développement à l'interne ou achat des sous systèmes*
 - *(*)Description technique du système*

Etape 2 ... Plan de Gestion Intégré (PGI) (suite)

- *PGI (suite)*
 - *(*)Un Work Breakdown Structure ([WBS](#))*
 - *(*)Identification des coûts de fabrication*
 - *Budget préliminaire, liste des sources de financement*
 - *Un plan d'utilisation des ressources en personnel*
 - *Prévisions préliminaires des besoins en personnel (en mois-homme)*
 - *Identification des ressources critiques*
 - *Identification des rôles*
 - *Identification des ententes de formation des équipes*
 - *Une liste des hypothèses fondamentales*
 - *(*)Une liste des risques (risk register)*

Etape 2 ... Plan de Gestion Intégré (PGI) (suite)

- *PGI (suite)*
 - *Liste des interdépendances et des inter-relations entre les différents programmes en cours de développement*
 - *Un plan préliminaire de support au développement:*
 - *Plan de gestion des documents (Configuration Management Control Plan)*
 - *Plan d'assurance qualité*
 - *Plan de logistique et de support sur le terrain (field support)*
 - *Plan de gestion des dépenses*
 - *Plan de mesures environnementales et de santé/sécurité*
- *Le PGI est un document d'environ 7 à 10 pages*

Etape 3

Développement du *baseline* du système

Etape 3: Définition du baseline du système

- Le *baseline* du système comprend les éléments suivants:
 - (*) *Diagramme des fonctionnalités* du système
 - (*) *Diagrammes-P*
 - (*) Liste de *compromis* (Matrice de Pugh)
 - *Spécifications* pour le système et *arbre des spécifications*
 - Spécification de *l'interface* du système
 - Modèle de *simulation* du système
 - (*) *Diagramme* du système physique
 - (*) *Plan de repli* face aux risques (registre de risque et plan de réaction face au risque)
 - Base de données de « *tracabilité* » des « requirements »
 - Liste des *technologies critiques*

Etape 3... Définition du baseline du système (suite)

- Le baseline du système comprend les éléments suivants (suite):
 - Identification des *fournisseurs principaux*
 - Identification des *installations principales* (et/ou critiques) de *test* et de *fabrication*

Etape 3...(suite)

- Diagramme des *fonctionnalités*
 - Illustre la *segmentation* du système en *fonctionnalités* sans considérer l'implantation physique des fonctionnalités
 - Les fonctionnalités sont les *actions* que le système doit prendre pour rencontrer les *exigences* (« requirements ») et pour produire la *réponse* désirée aux excitations d'entrée
 - Les fonctionnalités n'occupent aucun volume, n'ont aucune masse et ne sont pas « visibles »
 - Elles commencent par un *verbe d'action* suivi de *noms*
 - Les fonctionnalités complexes sont décomposées en fonctionnalités plus simples

Etape 3...(suite)

- **Diagramme du système physique**
 - Les **fonctions** identifiées dans le diagramme des fonctionnalités sont accomplies par des **dispositifs physiques** réels
 - Les **composantes hardware/software complexes** sont composées d'un **assemblage** de composantes plus simples
 - Un **module** (ou un sous-système) du diagramme peut implanter une ou plusieurs fonctionnalités ou seulement une fraction de fonctionnalité

Etape 3...(suite)

- La **conception** (design) du système consiste à effectuer un **mapping** des **fonctionnalités** («qu'est-ce que doit accomplir le système »...what do I need to do) vers les sous-systèmes formés de **dispositifs** hardware/software permettant d'implanter celles-ci (« que faut-il pour accomplir ces actions » ... what do I need to do it)
- Une **matrice** est utile pour illustrer le mapping entre les fonctionnalités et les sous-systèmes et leurs dispositifs physiques
- Un mapping 1 → 1 entre les fonctionnalités et les sous-systèmes physiques est hautement souhaitable parce que cela:

Etape 3...(suite)

- Simplifie la préparation des *spécifications*
 - Simplifie les procédures de *test*
 - Simplifie la spécification et l'implantation des *interfaces*
 - Facilite la *mise-à-niveau* (« upgrade ») du système
 - Facilite le *découpage du travail*
- (*)Exemple de diagramme des fonctionnalités pour le système d'alarme
 - (*)Exemple de diagramme physique pour le système d'alarme

Etape 3...(suite)

- **Guide** de préparation des spécifications. Une spécification définit ce qui doit être atteint en termes de paramètres critiques du systèmes:
 - Définir **l'utilité** de la spec. Et celui vers qui elle est destinée
 - Suivre les **standards** appropriés (MIL-STD-490 hardware, MIL-STD-498 software)
 - Préparer **l'arbre des spécifications** avant d'écrire les specs.
 - Permet d'établir un plan de ce que l'on a à faire
 - Assure un flux logique des specs
 - Réfléchir attentivement à la **spécification** et à sa **portée** (scope)
 - Disposer d'un **plan de contrôle de configuration** (configuration control)
 - Préparer le **format** (template) des spécifications *avant* de les écrire

Etape 3...(suite)

A faire lors de la préparation des spécifications

- Définir **l'utilité** de la spec
- Avant d'écrire une spec:
 - Raffiner le modèle des spécifications au besoin
- Préparer **l'arbre des spécifications** du projet
- Nommer un **responsable** du document des spécifications
- Définir un **plan d'écriture** des specs
- Inclure les **éléments** suivants dans les spécifications:
 - Interfaces
 - Diagramme physique
 - Diagramme fonctionnel
 - Description des modes d'utilisation du système
 - Matrice de vérification
- Inclure à la fois les exigences **fonctionnelles** et les exigences **physiques**
 - Utiliser **TBR** (To Be Revised) quand une spec. est préliminaire (ex. le poids de la brocheuse ne devrait pas dépasser 0,5 Kg)

Etape 3...(suite)

- Inscrire **TBD** (To Be Defined) quand la valeur de la spec est inconnue et qu'une estimation ne peut être faite lors de l'écriture (exemple: couleur du revêtement du boîtier du système d'alarme)
- **Minimiser** le nombre de TBR et TBD
- Inclure les **limites** et le **domaine de variation** de la spec (ex. le système d'alarme doit pouvoir fonctionner entre +40 et -40 degrés Celsius)
- Ecrire les spécifications clairement
 - Toutes les exigences doivent être nécessaires
 - La spec doit définir ce qui doit être atteint et non comment l'atteindre
 - Devrait être indépendante des autres specs
 - Décrire la spec en un seul paragraphe
 - **Le système doit être en mesure d'atteindre la spec**
 - La spec doit pouvoir être **vérifiée**

Etape 3...(suite)

- Les spécifications portant sur les performances obligatoires doivent utiliser le mot « **doit** » (shall) dans leur définition
- Les spécifications portant sur les performances non-obligatoires doivent utiliser le mot « **devrait** » (should) dans leur définition
- Le mot « **devra** » (will) ne doit être utilisé que pour exprimer une déclaration, mais ne doit jamais apparaître dans la définition d'une spécification
- Les versions de la rédaction des spécifications devraient être soumises à des **révisions par les pairs**
- Les spécifications peuvent être rédigées pour les **sous-systèmes** (ou les **modules**), ce qui rend leur performances plus claires que si l'on tente de les déduire à partir des spécifications de chacune de leurs composantes

Etape 3...(suite)

- **GARDER LES SPECIFICATIONS SIMPLES...**elles ne sont pas une solution de design, mais désignent plutôt ce que le système doit atteindre comme performance pour répondre aux exigences.

Etape 3...(suite)

A NE PAS faire lors de la préparation des spécifications

- Ne pas rédiger les specs avant que:
 - L'analyse soit complétée sur l'item pour lequel la spec est rédigée
 - Le rôle de la spécification soit compris par tous les intervenants
 - L'arbre des spécifications du projet soit complété
- Ne pas inclure de solutions de design dans la description de la spec
- Ne pas documenter inutilement
- Sans garder le bon niveau de contrôle de configuration
- Sans garder une « tracabilité » des specs

Etape 3 ...(suite) – La gestion du risque

- Eléments fondamentaux:
 - Comprendre ce qu'est le **risque**
 - Comprendre comment le risque doit être **abordé** (notion de **risk management**)
 - (*) Comprendre l'importance du **registre de risques** (**risk register**)

Etape 3 ...(suite) – La gestion du risque

- Qu'est-ce que le risque?
 - Probabilité qu'un événement **indésirable** se produise
 - **Conséquences** de l'occurrence d'un événement indésirable

Etape 3 ...(suite) – La gestion du risque

- Attitudes **dangereuses** face au risque:
 - **Ignorance** (jouer à l'autruche)
 - **Nier** son existence en adoptant des échéanciers optimistes
 - **Cacher des contraintes** (parfois planifiées, mais non diffusées aux membres de l'équipe)
 - Cacher le plus longtemps possible les **dépassements** de cédule
 - Faire de la **réétention d'information**

Etape 3 ...(suite) – La gestion du risque

- Attitudes **positives** face au risque:
 - Etre **honnête** et ne rien cacher
 - Etre **objectif** et utiliser des faits concrets plutôt que des impressions pour identifier les risques
 - **Précision**: vérifier les calculs d'estimation des risques
 - **Diffusion** de l'information et **justification** claire des décisions

Etape 3 ...(suite) – La gestion du risque

- Cinq facettes du risque à prendre en compte:
 - Aspects **techniques** relatifs à la performance du système face aux attentes du client
 - Aspects relatifs à **l'environnement du programme** de développement du système (ressources, clients, production, sous-contracteurs, fournisseurs)
 - Aspects relatifs au **support** du produit (entretien, formation, etc.)
 - Aspects relatifs au **coût**
 - Aspects relatifs à **l'échéancier**

Etape 3 ...(suite) – La gestion du risque

- Approches de prise en compte du risque
 - Utilisation d'un **registre de risques**
 - **Liste** des risques identifiés par le rapport **coût/bénéfice** associé à chaque risque
 - Stratégies de **réduction** des risques
 - Identification de l'entité **responsable** de gérer le risque jusqu'à son élimination
 - Associer un niveau de **priorité** à chaque risque
 - Etablir une liste de **critères** selon lesquels les décisions sur chaque risque sont prises
 - Le **chef de projet** crée et met à jour le registre de risques
 - La stratégie de **réduction** des risques doit faire partie du budget et de l'échéancier du projet

Etape 3 ...(suite) – La gestion du risque

- Approches de prise en compte du risque (suite)
 - Le **budget** doit prévoir un coût pour les risques afin de minimiser le coût global du projet en respectant les exigences de performance et l'échéancier
 - La liste des risques (stockée dans le registre de risques) doit être établie lors de la **préparation** de la proposition du projet (en réponse à un appel d'offres par exemple).
 - Le registre de risques ne doit pas seulement **énumérer** les risques, mais également les **décrire** en détails.
 - Il faut estimer le **niveau de vraisemblance** d'occurrence d'un risque et estimer les conséquences de sa présence en termes de **coûts directs et indirects**, de même qu'en termes de **délais** et de **dégradation des performances** du système
 - **Revoir** régulièrement la liste des risques et la mettre à jour

Etape 3 ...(suite) – La gestion du risque

Registre de risques

Type de risque	Niveau de priorité (1- faible, 5-élevé)	Conséquences de l'occurrence du risque	Coût en performance associé au risque	Probabilité d'occurrence	Coût estimé du risque (\$)	Plan de réduction du risque	Responsable du risque
Sensibilité des détecteurs de mouvement à l'éclairage ambiant	3	Fausse alarmes ou non-détection	Système non fonctionnel par temps ensoleillé	40%	3,000 \$	Possibilité d'ajouter un filtre IR au détecteur moyennant un coût de fabrication additionnel	D. Lauredeau
Présence d'animaux domestiques à l'intérieur de la maison lorsque le système est amorcé	4	Nombreuses fausses alarmes	Système non fonctionnel en présence d'animaux domestiques	55%	5,000 \$	Choisir un détecteur de mouvement dont on peut ajuster le seuil de détection en fonction de la taille des objets mobiles	D. Lauredeau
Disponibilité de détecteurs de mouvement avec seuil ajustable	3	Impossibilité de traiter le risque 2	Système non fonctionnel en présence d'animaux domestiques	20%	5,000 \$	Possibilité de concevoir un circuit satellite d'ajustement de seuil	D. Lauredeau
Disponibilité des micro-contrôleurs 8051	2	Impossibilité de fabriquer le système	Performance nulle	15%	Totalité du contrat!	Accords de distribution privilégiée signés avec trois fournisseurs	D. Lauredeau
Possibilité d'une grève à l'usine de montage	5	Retard de production impossible à prédire	Pénalité 10,000\$/semaine de retard sur la livraison des premières 1000 unités	25%	10000\$ x nb semaines conflit	Possibilité de sous-contracter 1000 unités à XYZ Co. Possibilité de fabriquer l'ensemble de la production à l'usine de St-Clin	Joe Boss
...							

Etape 3 ...(suite) – La gestion du risque

- La matrice de **Pugh** est un bon moyen de réduire les risques.
- Pour construire la matrice de **Pugh**:
 - Un concept est choisi comme **référence** (benchmark)
 - Des **concepts alternatifs** sont comparés au concept de référence selon des critères clairement établis

Matrice de Pugh

Nombre entre
-5 et +5

<i>Fonctionnalité</i>	<i>Concept référence</i>	<i>Concept alternatif 1</i>	
Détecter une intrusion	Détecteur mouvement IR	Détecteur Mouvement laser	-1
	Détecteur ouverture magnétique	Détecteur ouverture mécanique	0
Déclencher une alarme	Sirène électrique	Haut-parleur de caisse de son	-1
Alimenter le système	Pile 9 volts	Secteur	1
<i>Avantage global</i>			-1

Etape 4

Développement des sous-systèmes (allocated baseline)

Etape 4 Développement des sous-systèmes

- Cette étape consiste à:
 - Développer le **partitionnement** optimal des sous-systèmes
 - **Définir** et **documenter** les exigences (requirements) requises pour chaque sous-système et chaque **interface** entre les sous-systèmes
 - Établir les **spécifications** pour chaque sous-système aux niveaux **matériel** et **logiciel**
 - Établir les **spécifications** des **interfaces** entre les sous-systèmes
 - Concevoir les **diagrammes-blocs** des sous-systèmes (matériel et logiciel (potentiellement en UML))
- Les principes vus précédemment pour la description du système complet s'appliquent à chaque sous-système

Etape 5

Définition des performances du système

Etape 5 Définition des performances du système

- Cette étape consiste à:
 - Développer des outils permettant de **prédire** les performances du système et des sous-systèmes critiques
 - Définir la **méthodologie** pertinente à la **vérification** de chaque performance du système
 - **Observer** le niveau de performance des paramètres critiques tout au long du processus de design (notamment lors de changements ou lors du choix de concepts alternatifs pour réduire les risques)

Etape 5 ... (suite)

- Cette étape est accomplie en:
 - Effectuant des **simulations** pour les sous-systèmes et le système complet
 - Développant et en tenant à jour une **matrice de vérification** des performances
 - (*) Identifiant et observant l'évolution des **mesures de performances techniques** (MPT) (Technical Performance Measures ou TPM)

Etape 5 ... (suite)

- Qu'est-ce qu'une **mesure de performance technique** (MPT)?
 - **Indicateur** clé des performances techniques du système ou de l'un de ses sous-systèmes
 - Valeur attendue
 - Marge de sécurité
 - Variations
 - Choisie en utilisant la matrice de **DPF**
- La MPT est **observée périodiquement** (par exemple sur une base mensuelle) pour en vérifier la variation au cours du processus de développement (voir exemple).

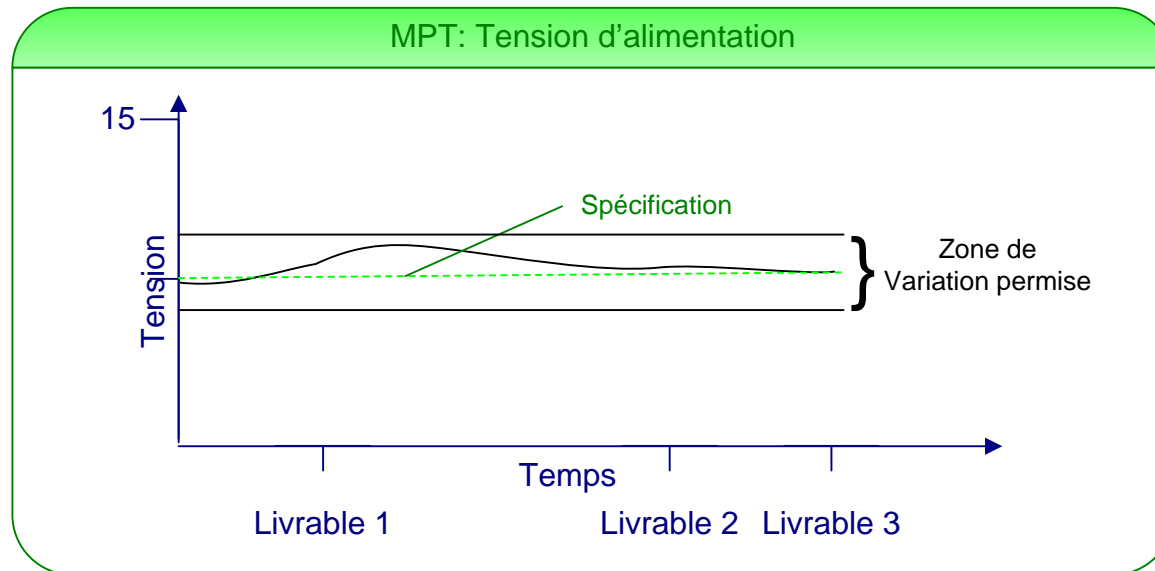
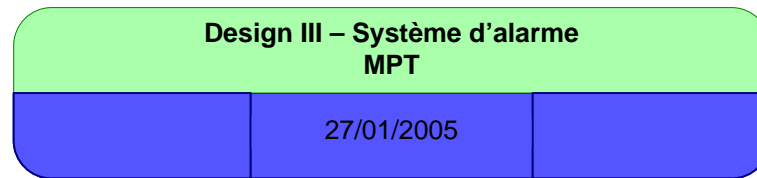
Étape 5 ... (suite)

- Avantages associés à l'utilisation des MPT:
 - Permet de **visualiser** l'historique des performances par rapport aux performances spécifiées
 - Permet la **prédiction** et la **détection** hâtives des problèmes
 - Permet de suivre les « **zones** » à **risque** du projet
 - **Documente** les principales hypothèses techniques et la direction que celles-ci prennent tout au long du développement du projet

Etape 5 ... (suite)

- **Méthodologie de génération des MPT:**
 - Identifier les paramètres critiques à suivre (à partir de la matrice DPF)
 - Identifier les spécifications et les valeurs-cibles pour chaque paramètre identifié
 - Déterminer les valeurs de performances présentes par modélisation, simulation, analyse, estimation ou mesure
 - Tracer l'évolution temporelle des paramètres pour analyser les tendances et prendre les actions nécessaires en cas de problème

Exemple de MPT pour le système d'alarme



Etape 6

Vérification des performances du système

Etape 6: Vérification

- Une fois les performances **définies**, il convient de **vérifier** si celles-ci respectent les exigences attendues du système (« requirement compliance »)
- Approche généralement adoptée:
 - (*) **Identifier** les exigences imposées pour les performances du système (« requirements ») en construisant une **matrice de vérification** (« Requirement Verification Matrix)
 - (*) Le **résultat** des vérifications est une **matrice de respect des exigences** (« Requirements Compliance Matrix)

Etape 6: Vérification (suite)

- (*) La matrice de **vérification des exigences** (Requirements Verification Matrix):
 - **Identifie** la **méthode** retenue pour vérifier chaque exigence (inspection, test, démonstration, analyse, différence/similitude)
 - **Identifie** le **niveau** pour lequel les vérifications seront effectuées (module, assemblage, sous-système, système)

Etape 6: Vérification (suite)

- Méthodes de vérification:
 - **Démonstration**: vérification de la fonctionnalité du matériel ou du logiciel **sans** l'aide d'outils de test (en faisant fonctionner le bidule)
 - **Inspection**: vérification **visuelle** de la forme, de l'ajustement et de la configuration du matériel ou du logiciel (regarder la pièce pour vérifier qu'elle est bien bleue)
 - **Similitude**: vérifie le respect des exigences en se basant sur l'utilisation garantie de **composantes similaires** dans des conditions identiques ou plus sévères (diode utilisée est la même que dans un design similaire)
 - **Test**: utilise des **techniques** (potentiellement sophistiquées) pour vérifier les performances (utiliser un interféromètre pour vérifier la courbure d'une surface)
 - **Analyse**: utilisation d'outils eux mêmes vérifiés pour vérifier les performances. Peut demander **l'analyse de données** expérimentales recueillies lors de mesures spécifiques (lors d'autres tests par exemple via une équation).

Etape 6: Vérification (suite)

- (*) Matrice de **vérification des performances**:
 - Liste les **performances** pour toutes les **exigences** (requirements)
 - Liste les performances du **simple** au **complexe** (c'est-à-dire composante->sous-système->système)
 - Identifie les **sources** des données de mesure des performances
 - Montre si le **design atteint le niveau de performance requis**
 - Montre tôt dans le processus de design si une performance n'est pas atteinte (non-compliance) et permet de concevoir un **plan** pour faire face à ce problème (mitigation plan).

Etape 6: Vérification (suite)

- **Objectifs** de l'étape de vérification:
 - Développer la **stratégie** et la **méthodologie** de test et les inclure dans le processus de design dès le début de manière à ce que les tests s'effectuent de manière optimale.
 - Développer les plans **d'intégration** et de **test** pour le système complet
 - Assurer que les **exigences principales** (« cardinal requirements) seront vérifiées via la matrice de vérification des exigences (« requirements compliance matrix »)
 - **Documenter** tous les tests importants des systèmes et sous-systèmes
 - **Analyser** les données de test

Etape 6: Vérification (suite)...planification des tests

- **(*)Planification** des **tests**...objectifs:
 - Définir le **processus** de test (« Test Plan »)
 - Développer les **équipements** de test en utilisant la même approche de design de système que pour la conception du système lui-même
 - Développer les **procédures de test** (« test procedure »), le processus **d'analyse des données** de test (nature, format, etc) et les **logiciels** d'analyse (type, nom, fournisseurs) en **parallèle** avec le design du système
 - Développer la **procédure de support** à l'équipe de test, l'approche d'intégration du système et les approches de **solution de problèmes**

Etape 6: Vérification (suite)...planification des tests

- Les étapes de planification des tests:
 - Identification des **objectifs** et définition des **priorités** afin de réduire l'impact potentiel de conflits de ressources
 - Identifier les **paramètres** sujets à vérification et nécessitant des mesures (MPT)
 - Identifier les besoins en termes de **données à recueillir** pour les tests
 - **Quantité** de données à recueillir en relation de la méthodologie de test (des données inutiles pour un test sans objet sont inutiles...De mauvaises données pour un test pertinent sont également inutiles)

Etape 6: Vérification (suite)...planification des tests

- Développer la **méthodologie** de test et les **procédures** de test:
 - **Documenter, valider, et diffuser** la méthodologie
 - Définir les **conditions initiales** d'exécution des tests, les conditions **d'interruption**, et les conditions **d'acceptation ou de rejet** des résultats des tests
 - Réduire la dépendance de la procédure de test par rapport au reste de la documentation (la documentation de test devrait autant que possible être autosuffisante)
- Préparer le **plan d'analyse** des données
 - Déterminer la **quantité** et le **type** de données à recueillir
 - Définir les **procédures d'analyse** de même que les **outils** qu'elles requièrent
 - **Valider** les procédures et les outils avant leur utilisation

Ceci peut avoir des conséquences sur les paramètres à mesurer (par exemple, il se peut qu'un test soit impossible à effectuer parce que le paramètre d'intérêt soit très difficile à mesurer...(ex. Columbia 01/02/03)

Etape 6: Vérification (suite)...planification des tests

- **Produits** résultant de la planification des tests:
 - **Méthodologie** de test du système (**comment** le système sera-t-il testé)
 - **Plan de test** du système (« Test Plan ») (**que** veut-on tester dans le système)
 - **Matrices de vérification**
 - **Liste** et des équipements de test de même que leurs **spécifications**
 - **Gabarit** pour le rapport des tests

Etape 6: Vérification (suite)...planification des tests

- **Avantages** offerts par la **planification** hâtive des tests:
 - Permet **d'améliorer** le design pour **faciliter** les tests
 - Permet **d'intégrer** les tests à la procédure de **vérification**
 - Permet d'identifier les **montages** de même que les **outils logiciels** requis pour effectuer les tests
 - Permet de définir la **séquence des tests** en fonction de la **séquence d'implantation du hardware et du logiciel**
 - Permet de **réduire** les risques

Etape 6: Vérification (suite)...planification des tests

- Remarques générales sur les tests
 - La **planification** des tests devrait faire partie intégrale des plans de développement d'un produit
 - La planification des tests devrait être faite par des membres appartenant à **toutes les disciplines**
 - Des tests appropriés pour les **sous-systèmes** favorisent le test du système intégré
 - Les **équipements** de tests devraient faire l'objet d'une **documentation** complète
 - La conduite des tests est un élément important du **budget** pour les **systèmes complexes** (par exemple les systèmes pour l'exploration spatiale)

Exemple de plan de test pour le système d'alarme

Niveau	Sous-niveau	Exigence	Méthode de vérification	Équipement requis	Méthode d'analyse
Détecter intrusion	Détecter mouvement	Vmin	Test	Chronomètre	Mesurer la vitesse avec le chronomètre et le ruban. Mesurer le signal à la sortie du
				Ruban à mesurer	
				Source d'alimentation	
		Détecteur XP-112			
		Oscilloscope numérique			
		Un passant			
	Vmax	Test	Chronomètre	Mesurer la vitesse avec le chronomètre et le ruban. Mesurer le signal à la sortie du	
			Ruban à mesurer		
			Source d'alimentation 5 v		
Taille	Test	Détecteur XP-112	Mesurer dimensions		
		Oscilloscope numérique			
Détecter ouverture	Ouverture min	Test	Un passant	Déplacer une partie du détecteur avec la vis en laissant l'autre fixe. Noter la tension aux bornes du	
			Règle		
			Détecteur Contact XP-23		
			Source alimentation 5 v		
			Vernier de déplacement linéaire		
Amorcer/Désamorcer...				Système de fixation du détecteur et du vernier	
				Spreadsheet	

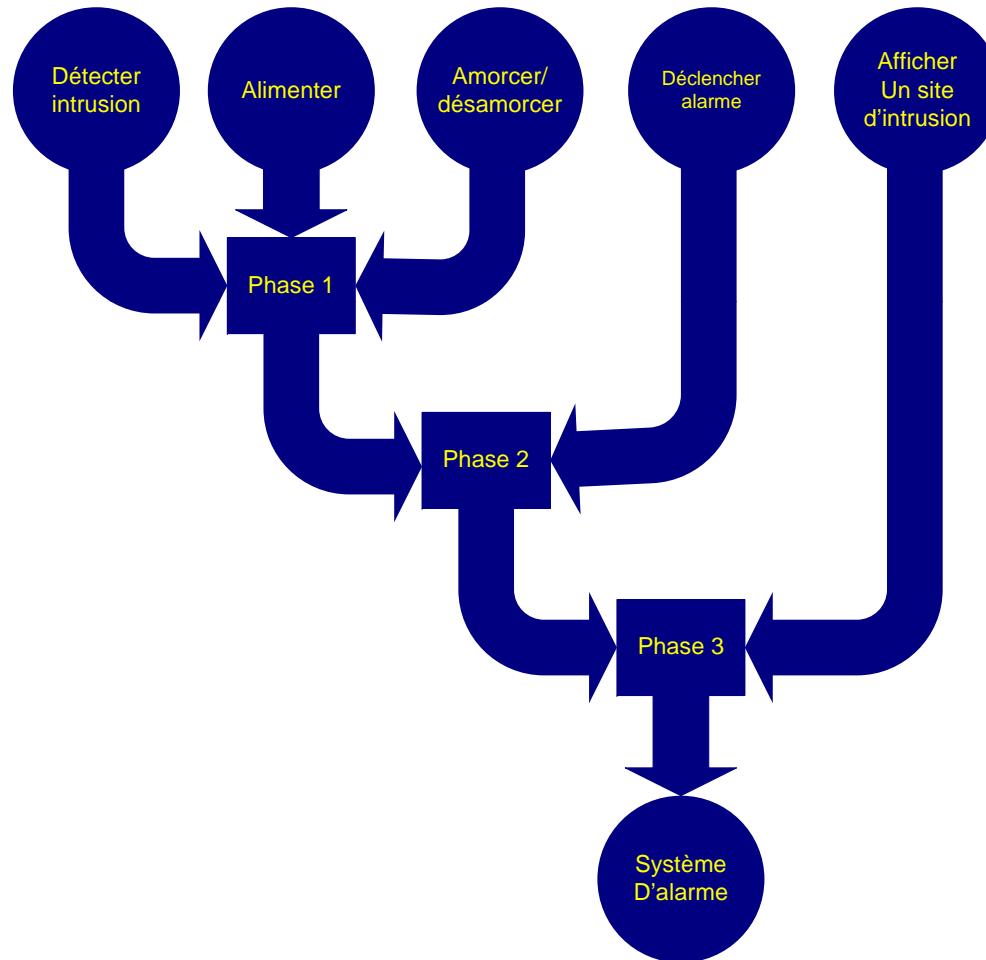
Etape 7

Intégration du système

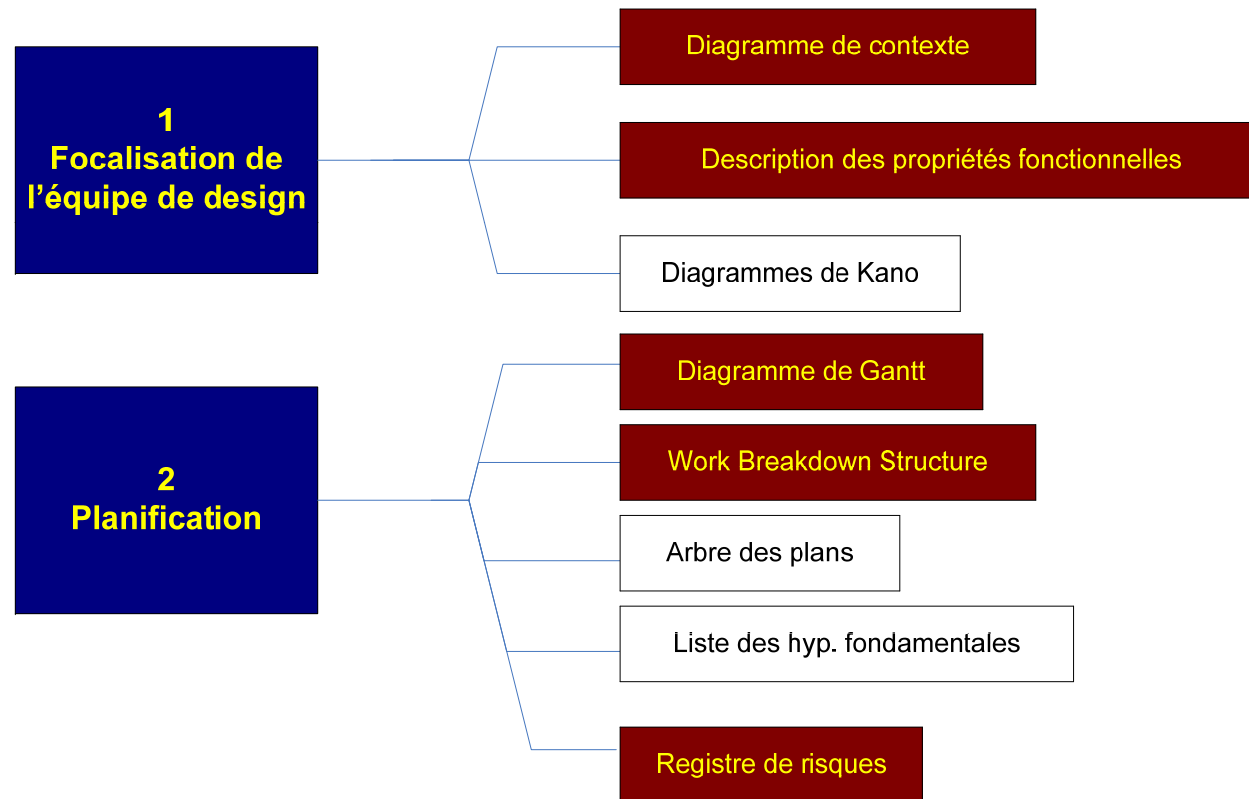
Etape 7 Intégration du système

- (*)Le plan d'intégration sert de guide à l'intégration des différentes parties (sous-systèmes du diagramme physique) du système complet
- Le plan d'intégration définit les étapes d'intégration et leur ordonnancement temporel
- Le plan d'intégration, dont le déroulement fait partie intégrante du diagramme Gantt, est conçu tôt dans le processus de développement afin de guider l'équipe et de permettre un suivi serré de étapes de design et d'implantation
- Le plan d'intégration doit être cohérent avec le plan de test
- L'intégration des sous-systèmes doit se faire le plus tôt possible afin de favoriser une étape de test rigoureuse et complète

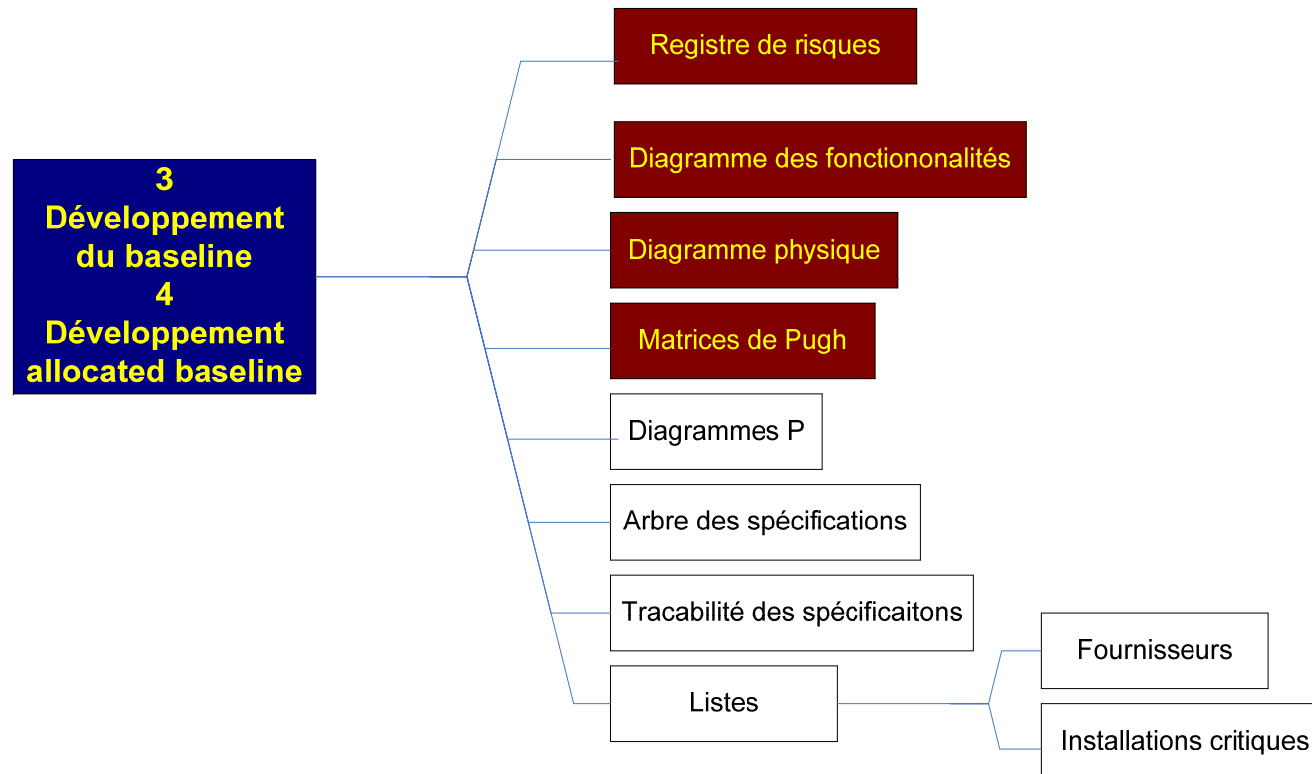
Etape 7 Intégration du système – Exemple plan d'intégration



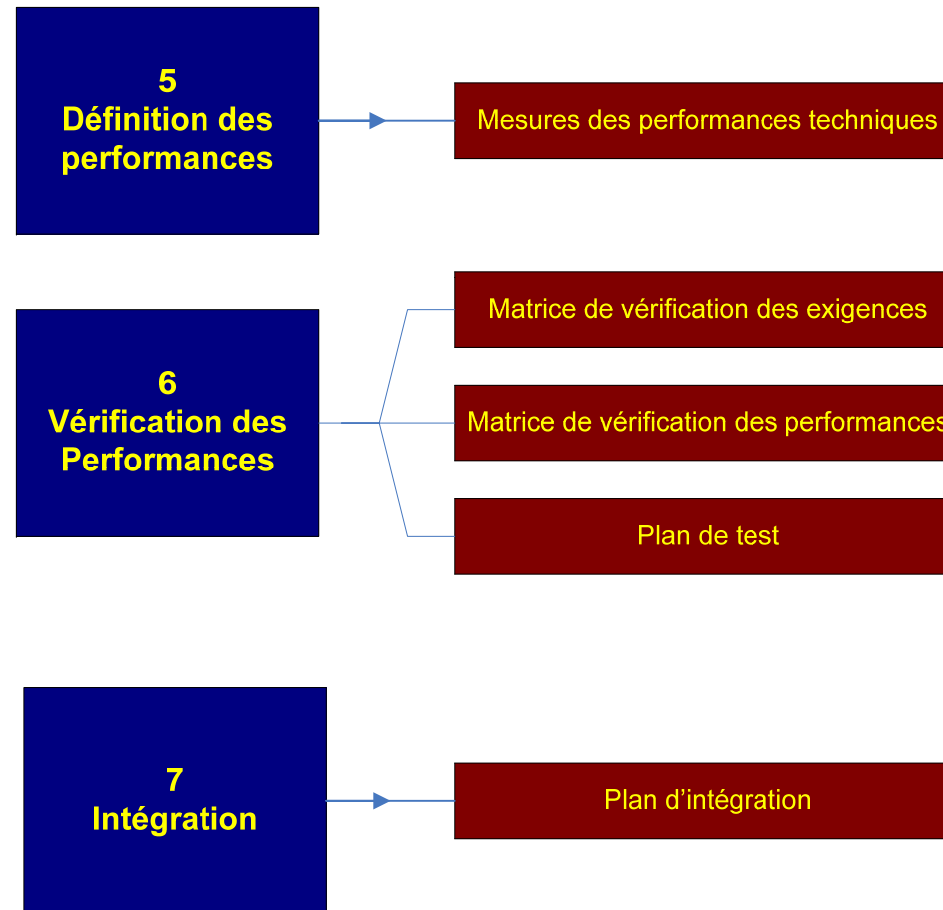
Les étapes du processus de développement (en bref - 1)



Les étapes du processus de développement (en bref - 2)

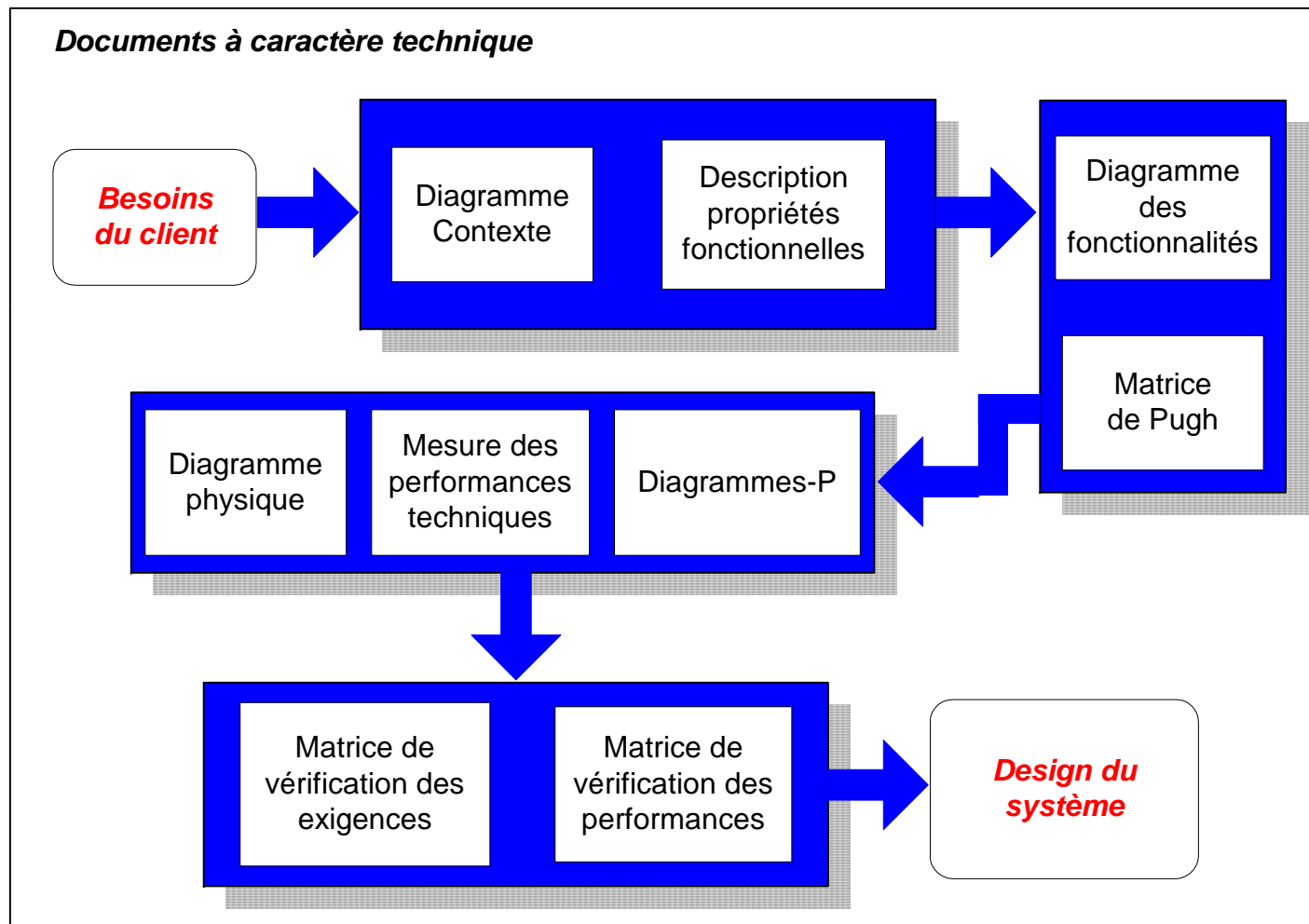


Les étapes du processus de développement (en bref - 3)

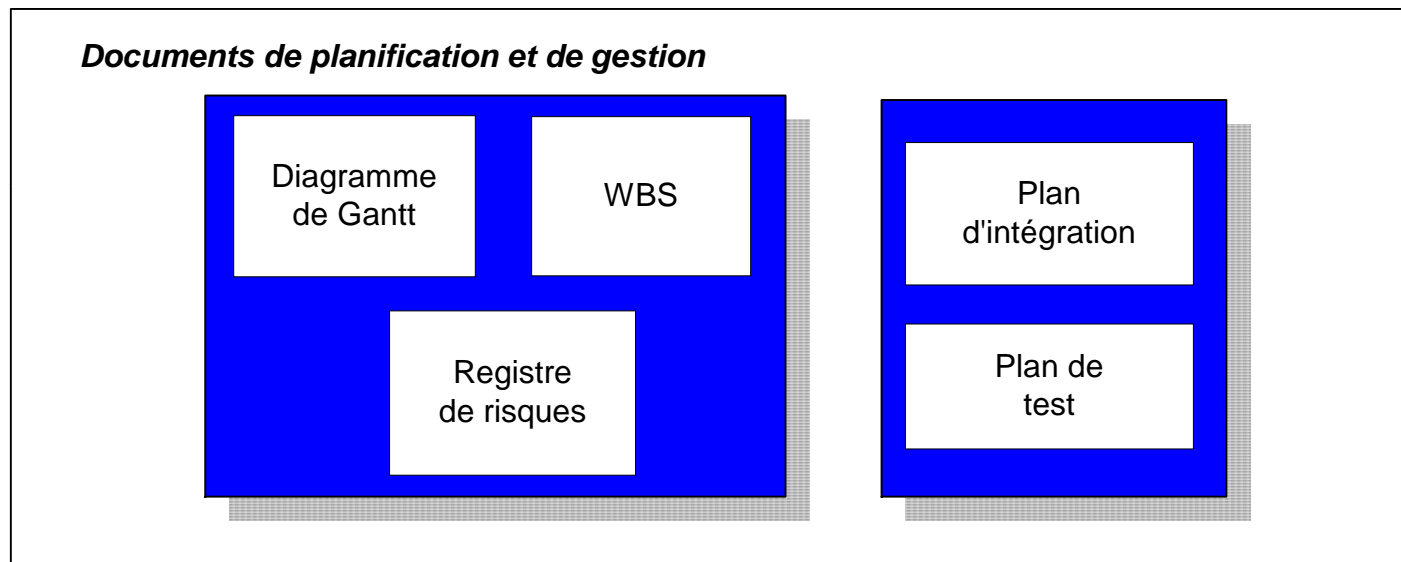


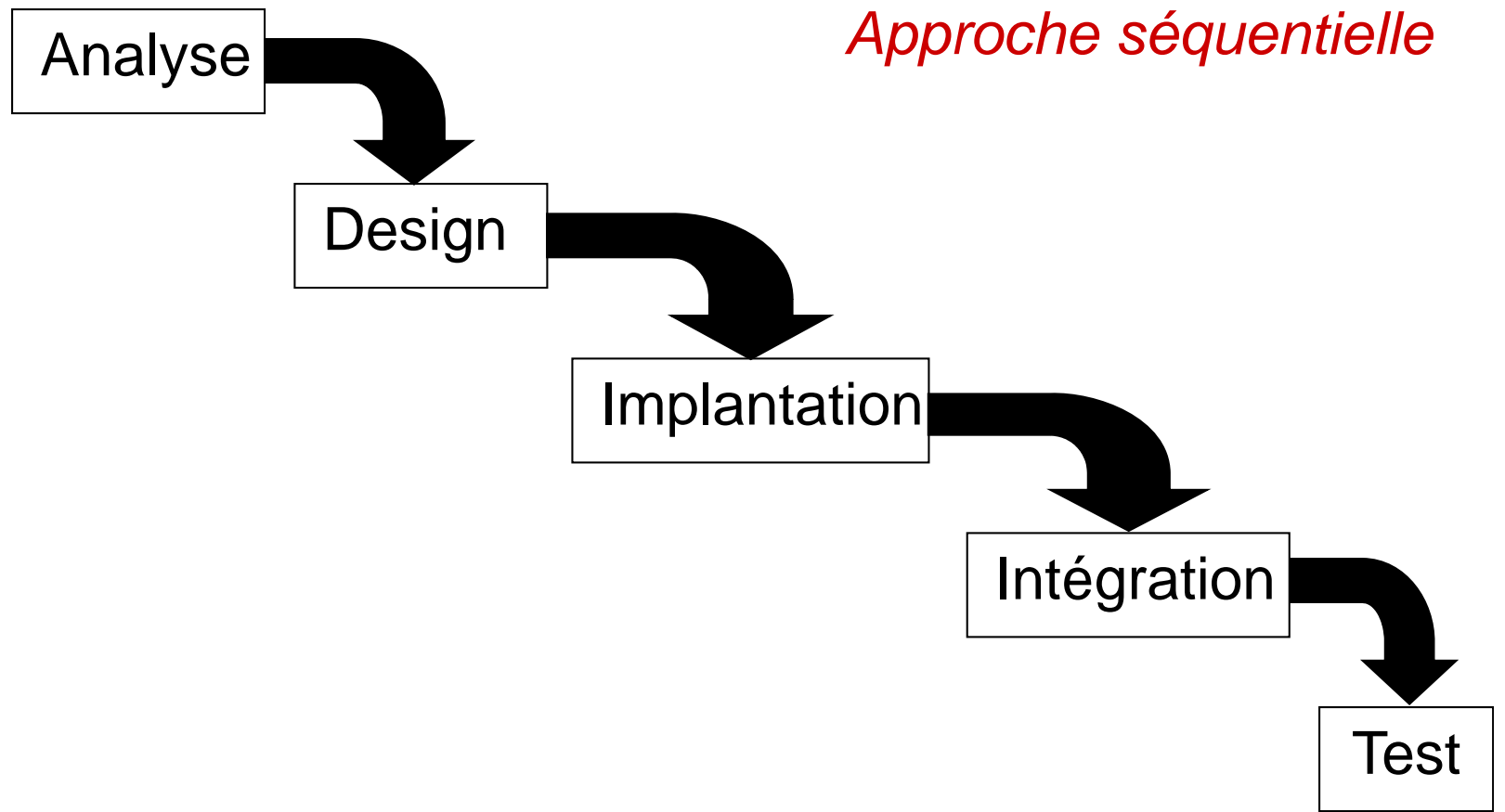
Documents importants du processus de développement

Documents importants...

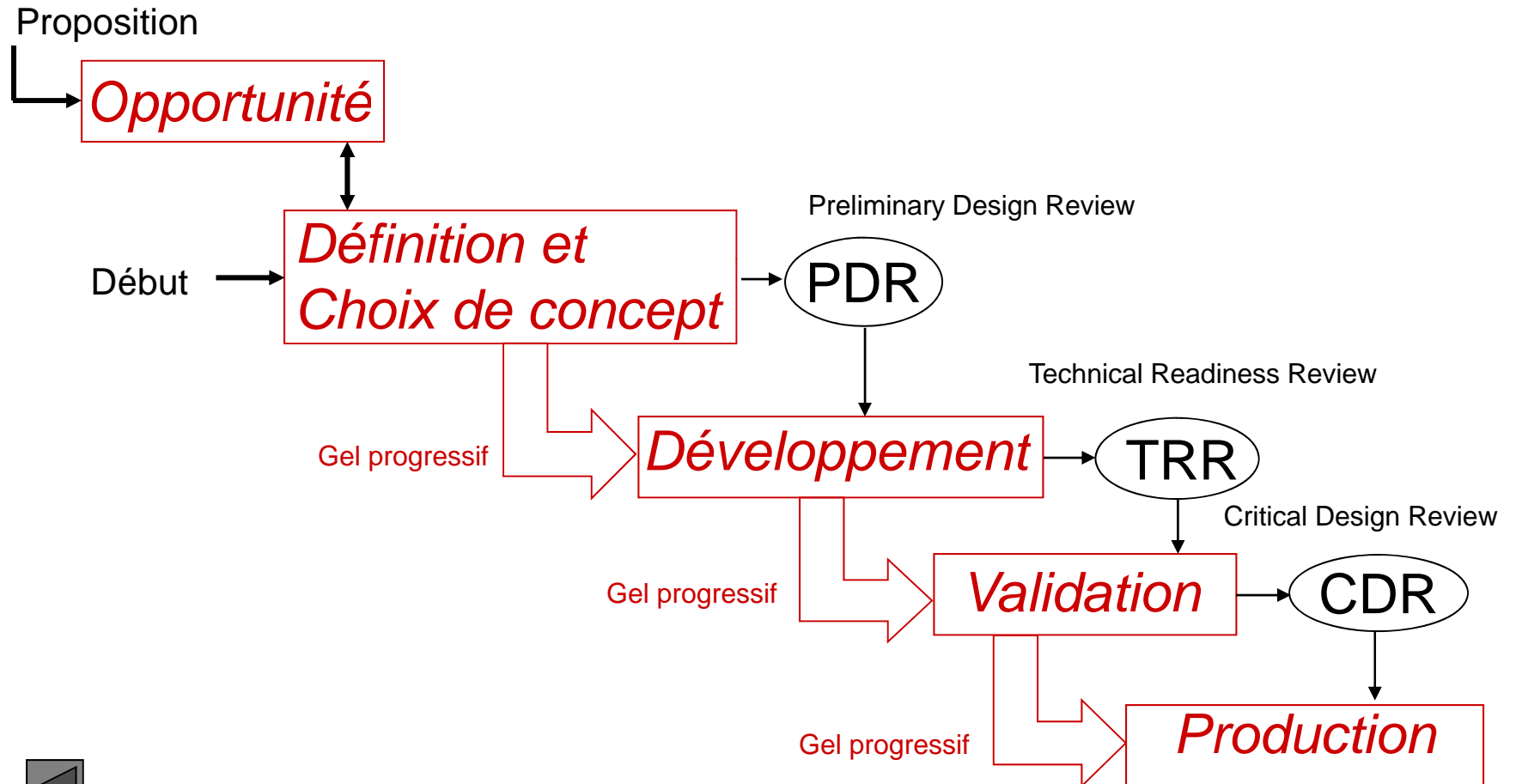


Documents importants...



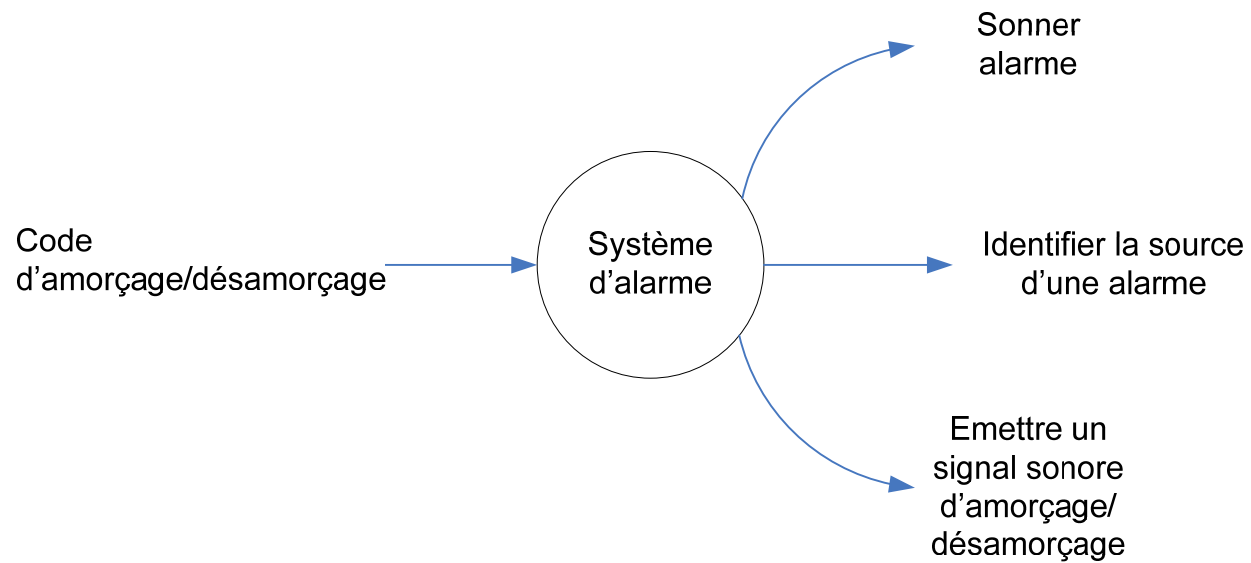


Approche d'ingénierie simultanée



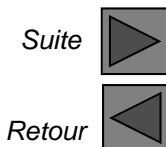
Design III – Système d’alarme – Diagramme de contexte

07/09/2007



DPF – Système d’alarme

Exigences du client	Fonctionnalités																	
	Détecter intrusion					Amorcer/désamorcer système			Afficher provenance	Déclencher alarme			Alimenter système	Respecter standards	Respecter faible coût	Respecter apparence	Respecter entretien minimum	
	Détecter mouvement			ouverture porte /		Délai	Code	Signal sonore	no	Intensité (db)	Temps réponse (sec)	Durée (sec)	Valim (volts)					
	Vitesse min (m/s)	Vitesse max (m/s)	Taille min (cm)	Ouv. Min (mm)	Taux dét. (%)													deltaT (sec)
Détecte ouverture porte				5	5						3		5					
Détecte ouverture fenêtre				3	5						3		5					
Détecte mouvement	5	5	5										5					
Amorçage avec code secret						4	5	3					3					
Désamorçage avec code secret						4	5	3					3					
Emission signal sonore amorçage								3					2					
Emission signal sonore désamorçage								3					2					
Déclenchement d'une alarme lors d'une intrusion											3	5	4	5				
Affichage de la source de l'intrusion									5				1					
Apparence discrète																5		
Facilité d'entretien													3				5	
Standards nord-américains													5	5				
Faible coût													4			5		



DPF – Système d’alarme, *matrice de couplage*

	Détecter intrusion	Amorcer/Désamorcer	Afficher provenance	Déclencher alarme	Alimenter système	Respecter Standards	Respecter faible coût	Respecter apparence	Respecter entretien min.
Détecter intrusion	5		2	3	5		4	2	3
Amorcer/Désamorcer		5					4	2	3
Afficher provenance	2		5	2			4	2	1
Déclencher alarme	3		2	5	5		4	3	3
Alimenter système	5	2	2	5	5	3	3		3
Respecter Standards					3	5			
Respecter faible coût	4	4	4	4	3		5	4	3
Respecter apparence							4	5	2
Respecter entretien min.	3	3	1	3	3		3	2	5



Etape 1 ... Construction de diagrammes de Kano

Diagramme de KANO – Système d’alarme

26/10/2004

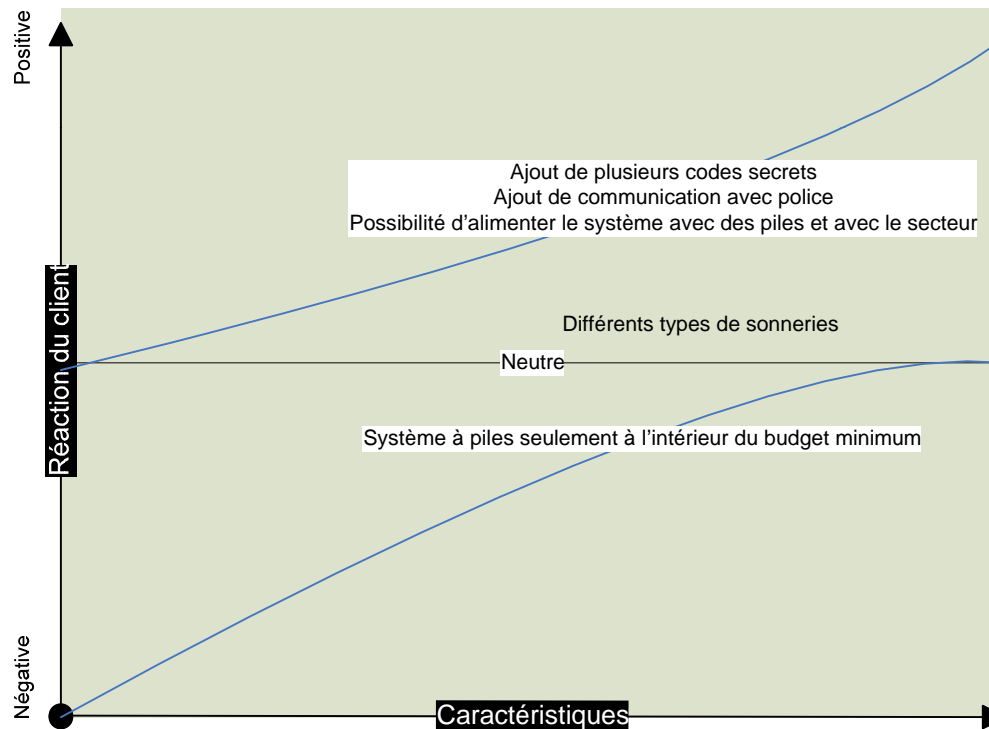


Diagramme de Gantt
Système d'alarme
Voir diagramme Visio



Le diagramme de Gantt

- Le diagramme de Gantt représente une description graphique:
 - Du déroulement des tâches du projet dans le **temps**
 - Des **liens** entre les tâches et des **contraintes** imposées par certaines tâches sur d'autres tâches
 - Des **ressources** affectées à l'exécution des tâches
 - Du **coût** associé à l'exécution des tâches



WBS – Système d’alarme

Voir diagramme Visio

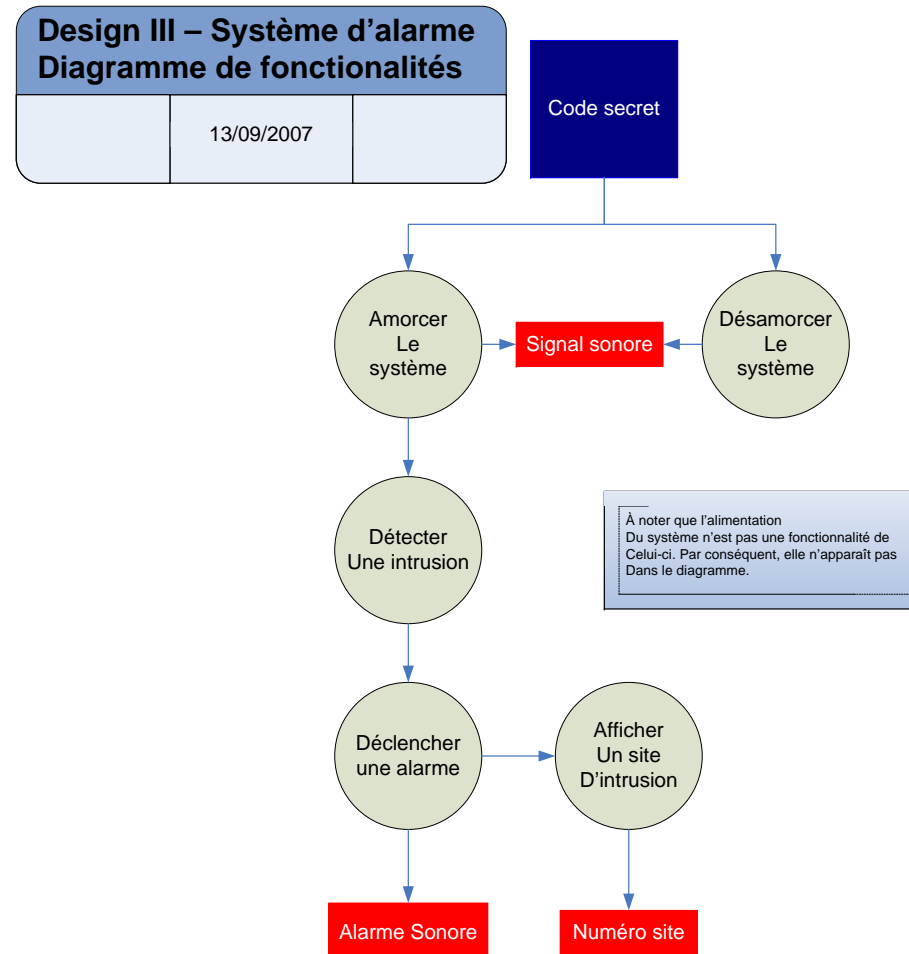


Le Work Breakdown Structure (WBS)

- Le WBS illustre de manière différente les tâches montrées dans le diagramme de Gantt
 - Les tâches sont regroupées en **classes** propres à différents types d'activités (rapports, design, etc)
 - Les tâches sont illustrées en termes de blocs de travail sans lien avec la distribution temporelle sur laquelle elles seront exécutées
 - En général, le client paie sur les tâches complétées (i.e. un **work unit** correspondant à une case du WBS), à condition qu'elles soient effectuées dans les temps prévus pour le projet



Diagramme de fonctionnalités



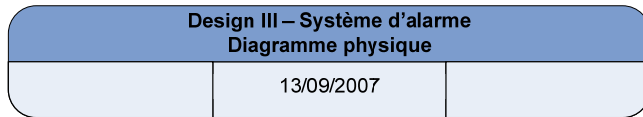
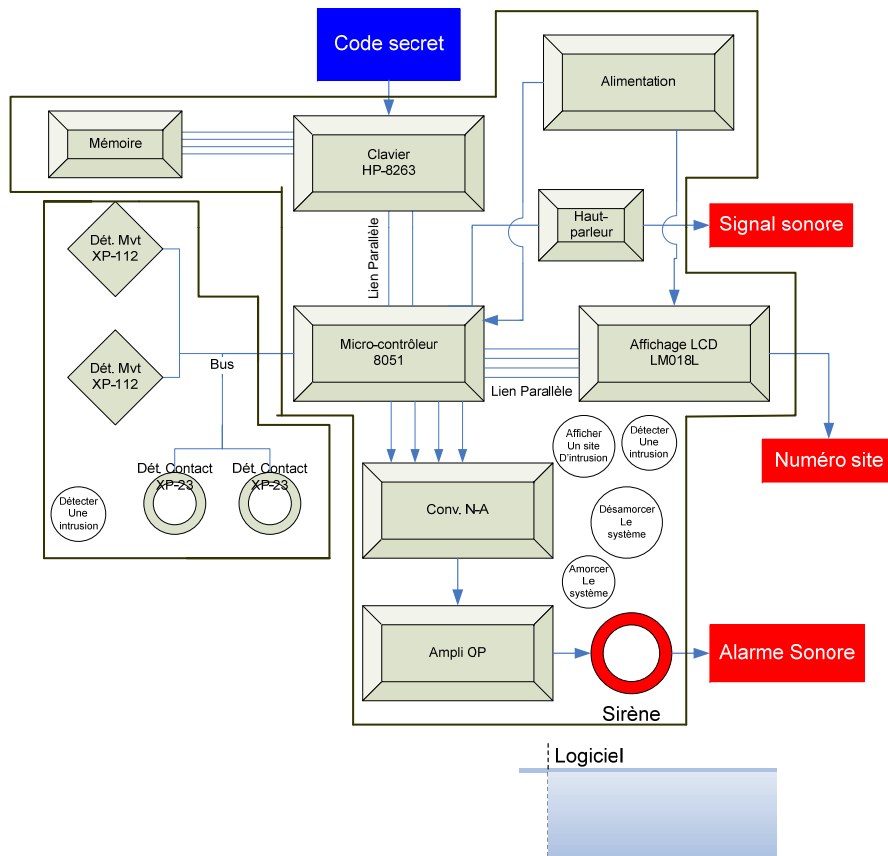


Diagramme physique



Exemple de matrice de vérification des exigences

Document	Niveau	Sous-niveau	Paramètre critique	NSP	Méthode de vérification	Commentaires
DPF	Détecter intrusion	Détecter mouvement	Vmin (m/s)		Test	
			Vmax (m/s)		Test	
			Taille min (cm cube)		Test	
		Détecter ouverture	Ouverture min. (mm)		Test	
			Taux détection (%)		Démonstration	Essayer 100 fois et voir le nombre de succès
DPF	Amorcer/désamorcer	Délai	deltaT (sec)		Test	
		Code	nbDigits		Démonstration	Faire fonctionner le système
		Signal sonore	Intensite (db)		Test	
DPF	Affichage source		no		Inspection	



Exemple de matrice de vérification des performances

Document	Niveau	Sous-niveau	Param. critique	NSP	Méthode de vérification	Spécification	Performance	Marge	Commentaires
DPF	Détecter intrusion	Détecter mouvement	Vmin (m/s)		Test	0.50+/-0.1	0.52+/-0.1	0.02	
			Vmax (m/s)		Test	1.5+/-0.1	1.6+/-0.2	0.1	
			Taille min (cm cube)		Test	12	10	2	
		Détecter ouverture	Ouverture min. (mm)		Test	1	1	0	
			Taux détection (%)		Démonstration	98%	99%	1%	
DPF	Amorcer/désamorcer	Délai	deltaT (sec)		Test	1.5	1	0.5	
		Code	nbDigits		Démonstration	4	4	-	
		Signal sonore	Intensite (db)		Test	10	9.8	-0.2	Malgré que la spécification ne soit pas rencontrée, on constate que l'alarme est audible de très loin et qu'elle est par conséquent suffisante
DPF	Affichage source		no		Inspection				Dans les cas détectés, le numéro est toujours exact

