

Le génie logiciel

Qu'est-ce que c'est le génie logiciel et sa place dans les sujets d'informatique

Le contenu est basé sur les transparents de la 10^{ème} édition de “*Software Engineering*” de Ian Sommerville

Génie logiciel



Logiciel (ou Programme)

- Programmes et la documentation associée – cahier de charges, modèles, manuels
- Types
 - Générique
 - Individuel
 - Hérité

Questions de l'ingénierie de logiciel

- Qu'est ce que c'est logiciel?
- Qu'est ce que c'est génie logiciel?
- Quelle est la différence entre génie logiciel et informatique?
- Quelle est la différence entre génie logiciel et génie des systèmes?
- Qu'est ce que c'est processus unifié de logiciel?
- Qu'est ce que c'est modèle du processus de logiciel

Questions de l'ingénierie de logiciel

- Combien ça coûte?
- Quelles sont les méthodes du génie logiciel?
- Qu'est ce que c'est CASE (Computer-Aided Software Engineering)
- Quelles sont les propriétés du bon logiciel?

Projets Génie logiciel

- **Succès**

.....

- **Mitigé**

.....

- **Échec**

.....

Projets Génie logiciel

- **Succès**
 - livré à temps, à l'intérieur des budgets et des spécifications originales
- **Mitigé**
 - livré et opérationnel mais avec moins de fonctions que prévu en dépassant le budget et/ou les échéanciers
- **Échec**
 - abandonné en cours de route, ou résultat livré mais jamais utilisé

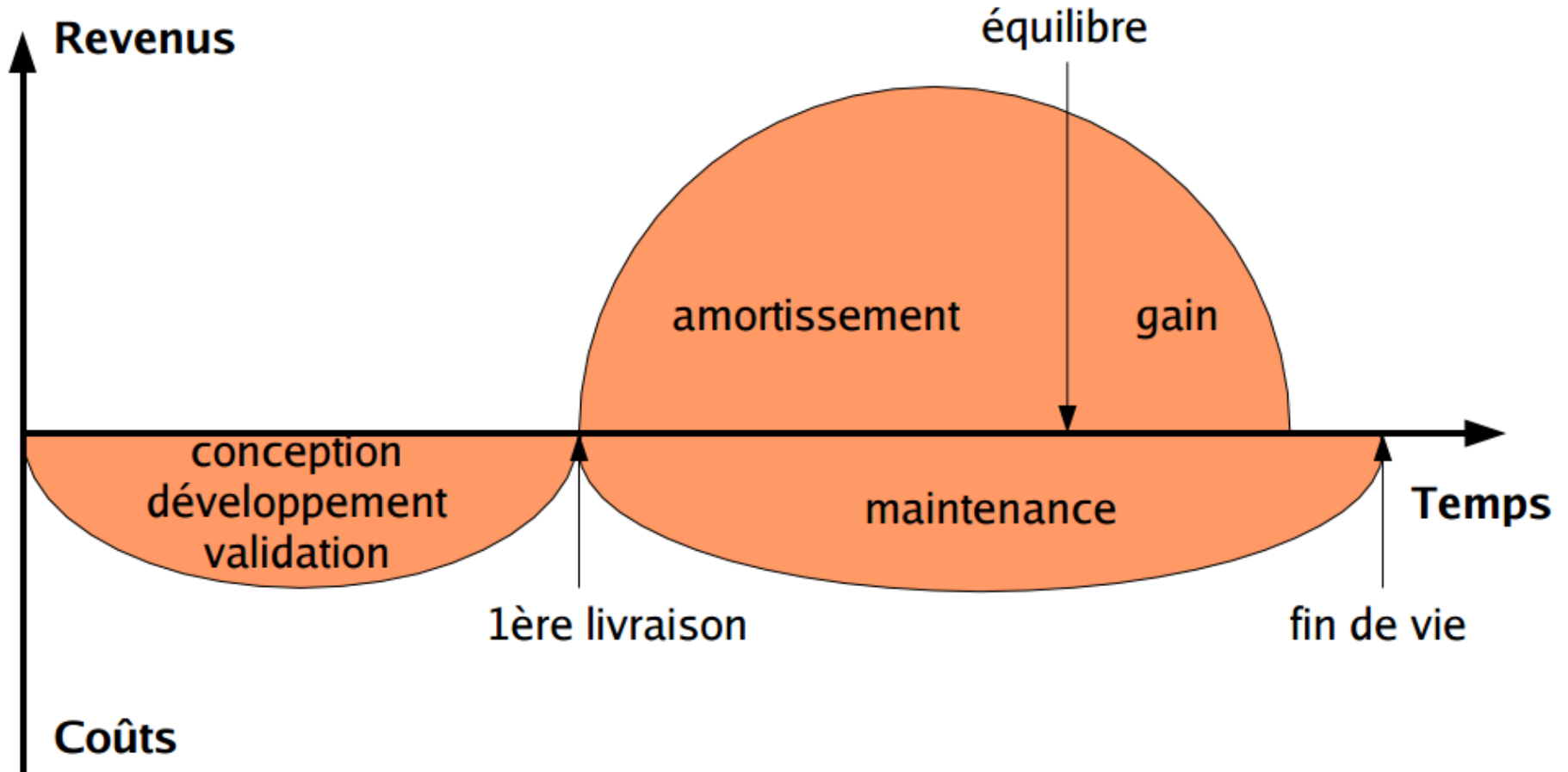
Processus du logiciel

- Un ensemble d'activités dont l'objectif est le développement et l'évolution du logiciel.
- Activités :
 - Spécification – ce que le logiciel doit faire et les contraintes posées au développement
 - Développement - production logiciel
 - Validation – vérification si le logiciel est celui qui est attendu du client.
 - Évolution – modification du logiciel en accordance avec les besoins.

Modèle du processus

- Le modèle – une présentation simplifiée d'un point de vue différent
- Points de vue:
 - Flux d'activités
 - Flux des données
 - Rôles/activités
- Modèles génériques
 - Cascade (Waterfall)
 - Itérative
 - Composants

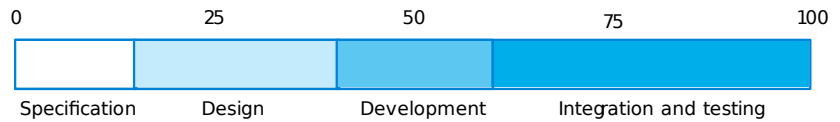
La vie du logiciel



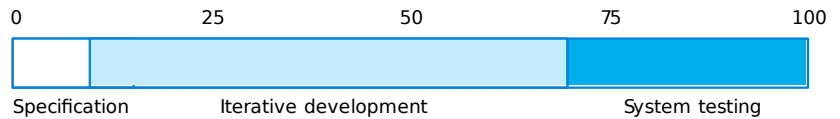
Les coûts

- Activités
 - Spécification du logiciel
 - Développement
 - Validation
 - Evolution

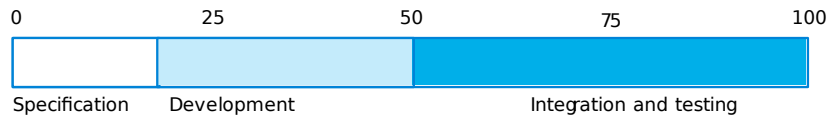
Waterfall model



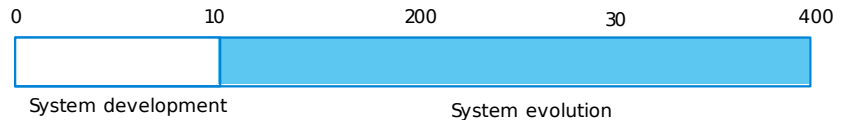
Iterative development



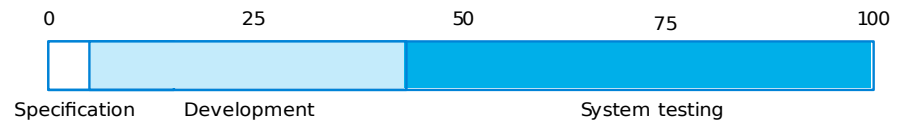
Component-based software engineering



Development and evolution costs for long-lifetime systems



- Production



Méthodes de génie logiciel

- Composants des méthodes
 - Modèles – graphiques (objets, flux des données, machine d'états et c.)
 - Règles - contraintes
 - Recommandations – bonne pratique
 - Direction et gestion – la séquence des activités

CASE (Computer-Aided Software Engineering)

- Upper-CASE
 - Support les activités de conception et de definition des besoins
- Lower-CASE
 - Support les activités tardives – programmer, déboguer, tester

Les propriétés du bon logiciel

- Avoir la fonctionnalité désirée.
- Facilement maintenu – de couvrir les besoins changés
- Sûr – on doit avoir confiance en lui
- Efficace – de ne pas gaspiller les ressources du système
- Accepté, compris par les usagers

Les défis devant le GL

- Hétérogénéité des plateformes
- Délivrance (respecter les termes et la qualité à la fois)
- Confiance des usagers
- Responsabilités professionnelles et éthiques

Les défis devant le GL

1

ce que la
direction
demande



2

ce que
demande
le chef de
projet



3

ce qui est
conçu



4

ce qui est
programmé



5

ce qui est
installé



6

ce que
l'utilisateur
souhaitait



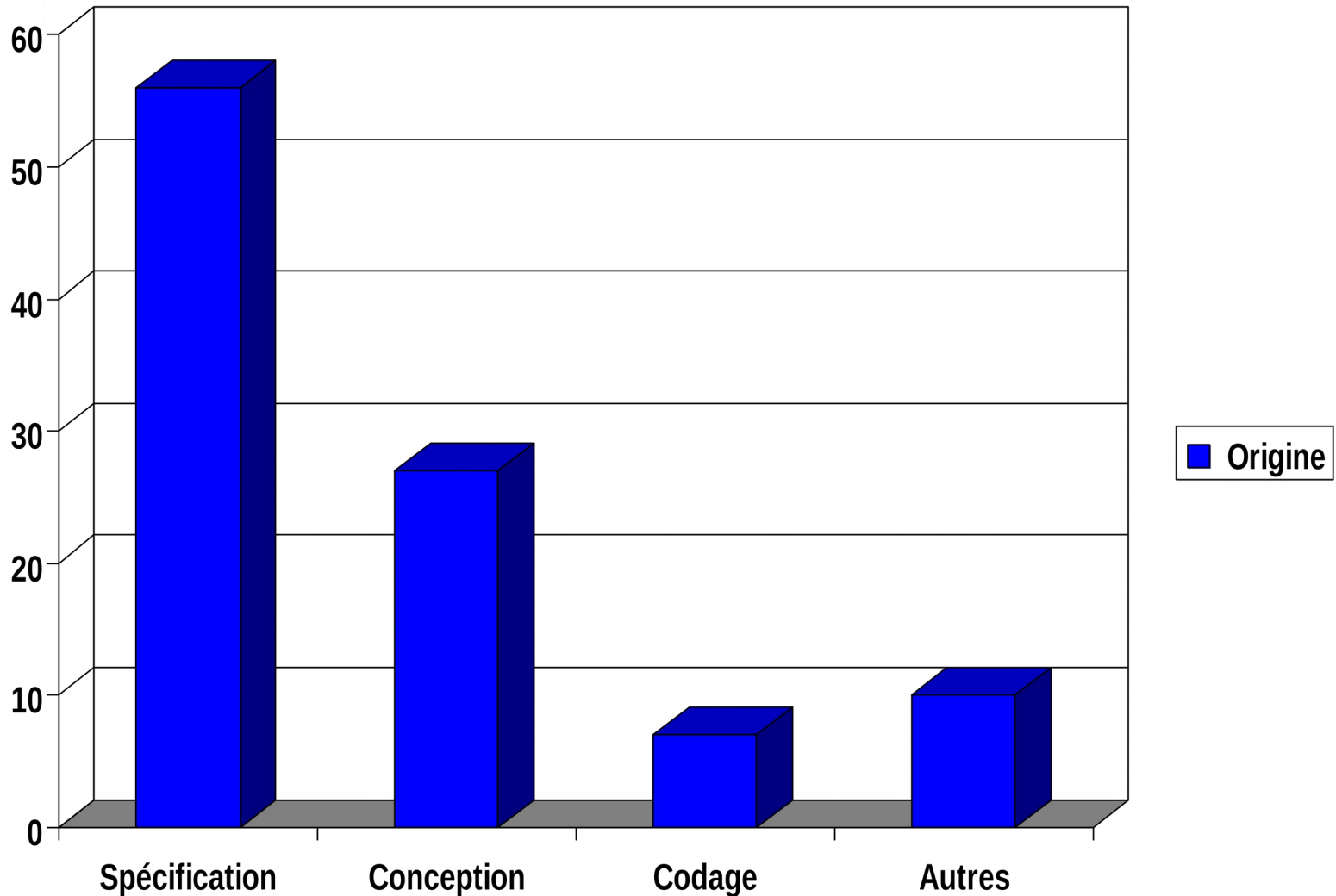
Systèmes critiques

- Système critique : sécurité
 - Perdre la vie ou la santé. Ex. Usine chimique
- Système critique : mission spatiale
 - Une activité essentielle est échouée - Navette Spatiale
- Système critique : finance
 - Grand pertes d'argent – système de comptabilité d'une banque

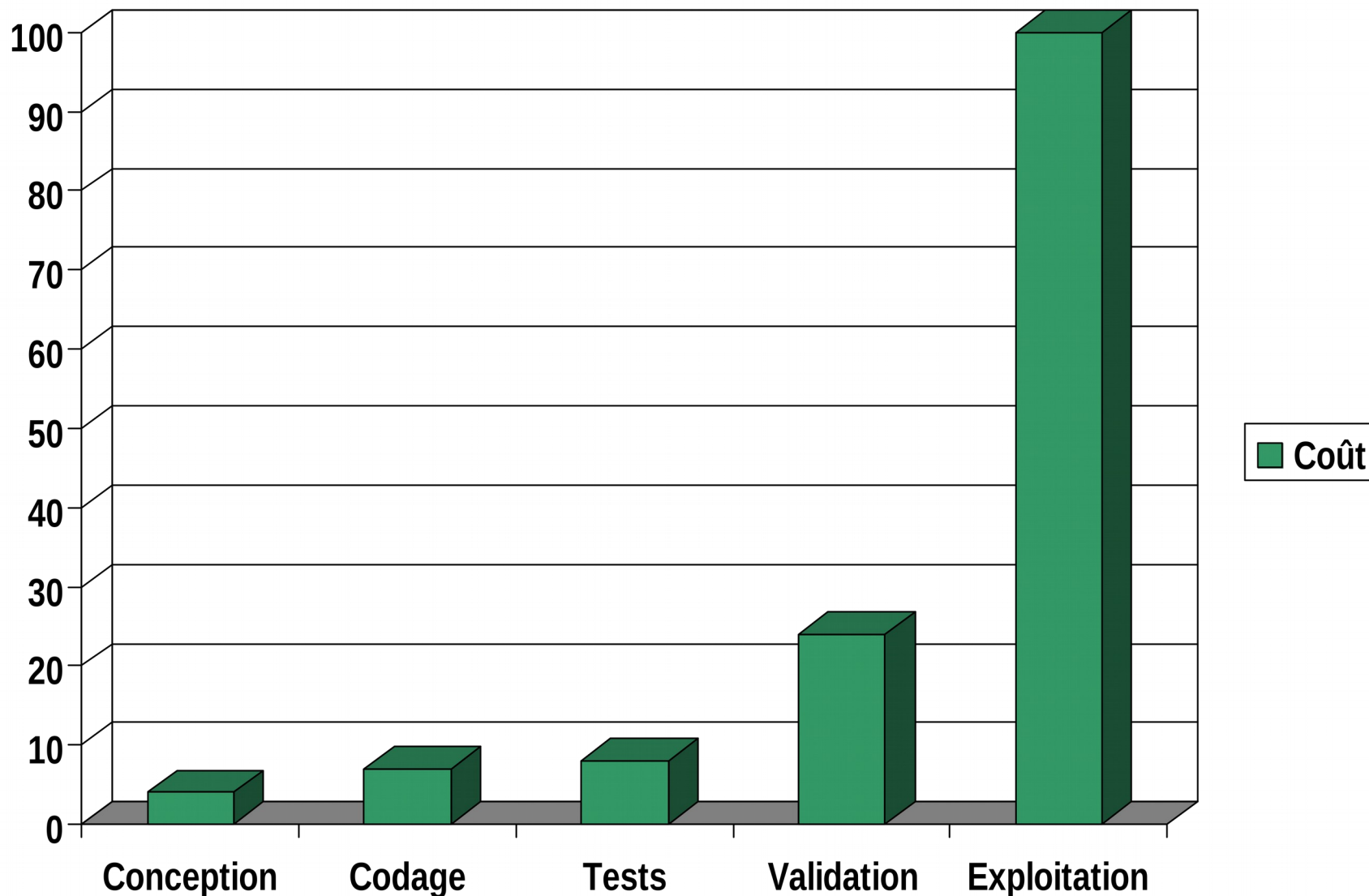
Fiabilité

- Panne du matériel
- Échec du logiciel
- Erreur opérationnelle – le plus souvent

Fiabilité : Origine de l'erreur (logiciel)



Fiabilité : Coût de réparation

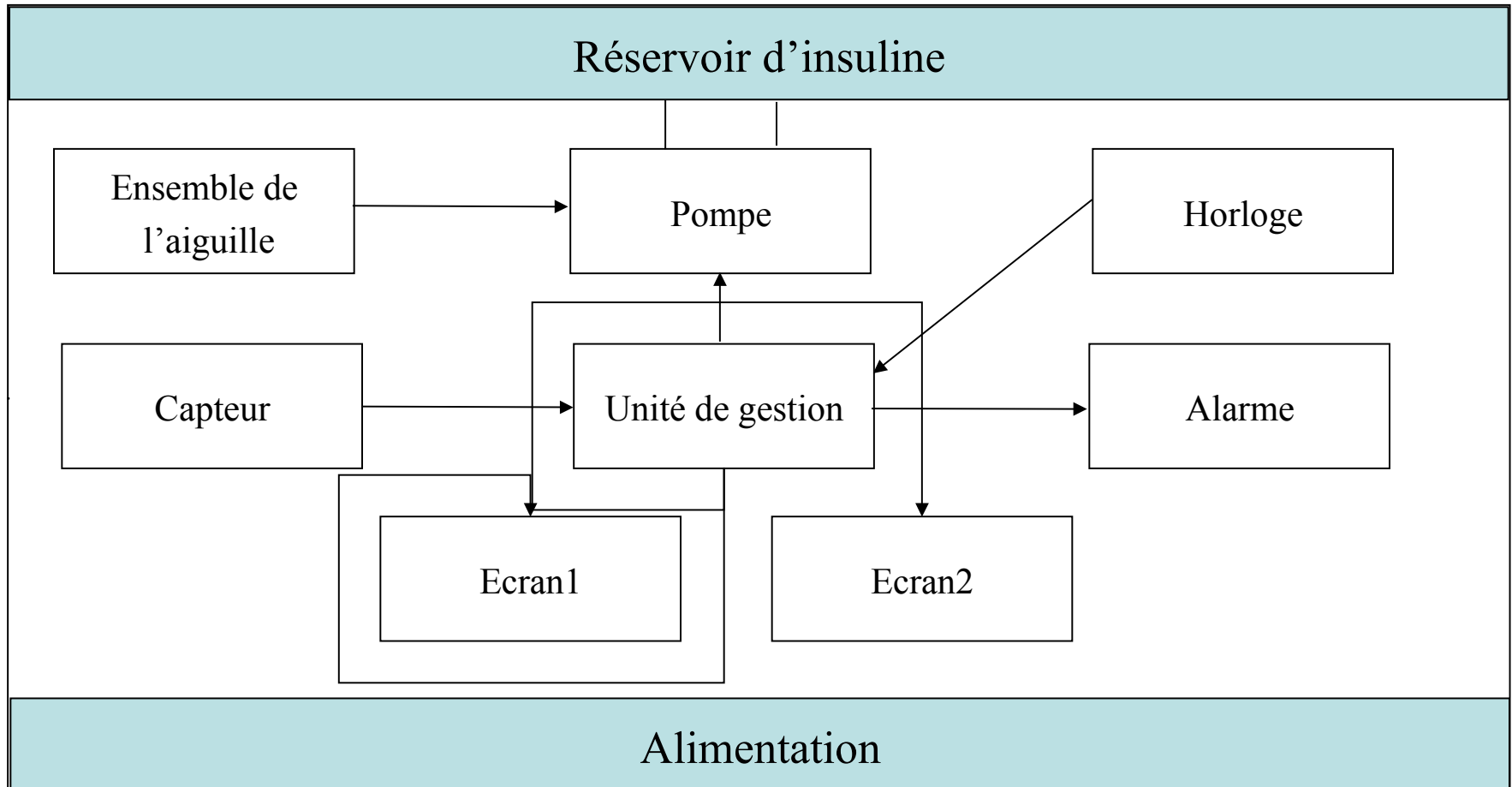


Fiabilité

- Composants de la fiabilité (Dependability)
 - Disponibilité
 - *Fiabilité (Reliability)*– l'habilité de assurer les services comme ils sont spécifiées
 - Sécurité
 - (Safety)De fonctionner sans échec catastrophique
 - (Security)De se protéger des attaques externes
 - Habilité de restauration après un échec
 - Habilité d'être maintenu – à quel degré il s'adapter vers de nouvelles exigences
 - Habilité de survivre – de quel degré il continue de fonctionner sous attaque
 - Tolérance d'erreurs – à quel degré il tolère les erreur de l'utilisateur.

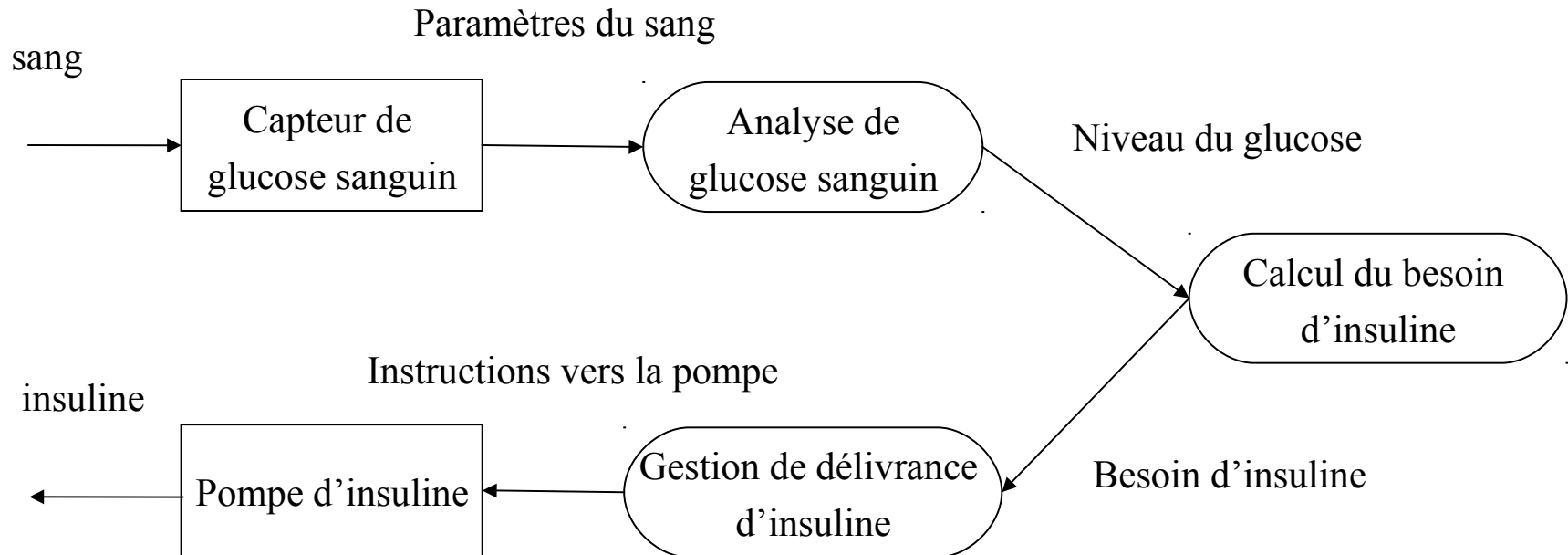
Exemple - Insuline pompe

- Organisation



Exemple - Insuline pompe

- Flux de données



Exigences de fiabilité

- Le système doit être capable de livrer l'insuline quand l'organisme a besoin et en quantité qui assez de neutraliser le glucose.
- L'exigence principale de sûreté et de ne pas livrer une surdose, qui peut être mortelle.

Les coûts de fiabilité

