

The logo of the University of Strasbourg, featuring two blue curved segments that form a stylized 'S' shape.

UNIVERSITÉ DE STRASBOURG

Rhénatic : théma-technique

La démarche Qualité dans le développement logiciel

4 décembre 2009 - Jean Martin

A thick orange horizontal bar at the bottom of the slide, with a rounded top-left corner.

Jean MARTIN : ma double vie (1/2)

- Chef de Projet :



Les bugs fâchent les
utilisateurs

O.k., maintenant tu fais ce
que je te dis !



Jean MARTIN : ma double vie (2/2)





- Enseignant associé Université de Strasbourg - IUT Robert Schuman

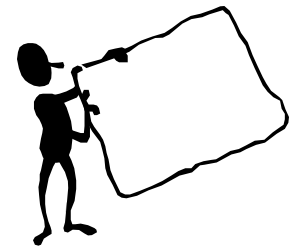


Rien n'arrête l'automatisation



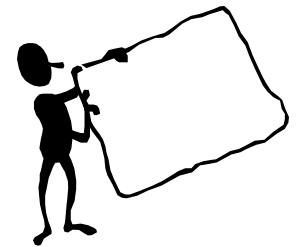
Plan de l'Intervention

-  **1. La problématique : ce n'est pas si simple**
-  **2. Les processus d'intégration et de qualification**
-  **3. La Qualité du logiciel**
-  **4. Et plus loin encore ...**



Plan de l'Intervention

1. La problématique : ce n'est pas si simple



Les ingrédients

Les acteurs

Les niveaux

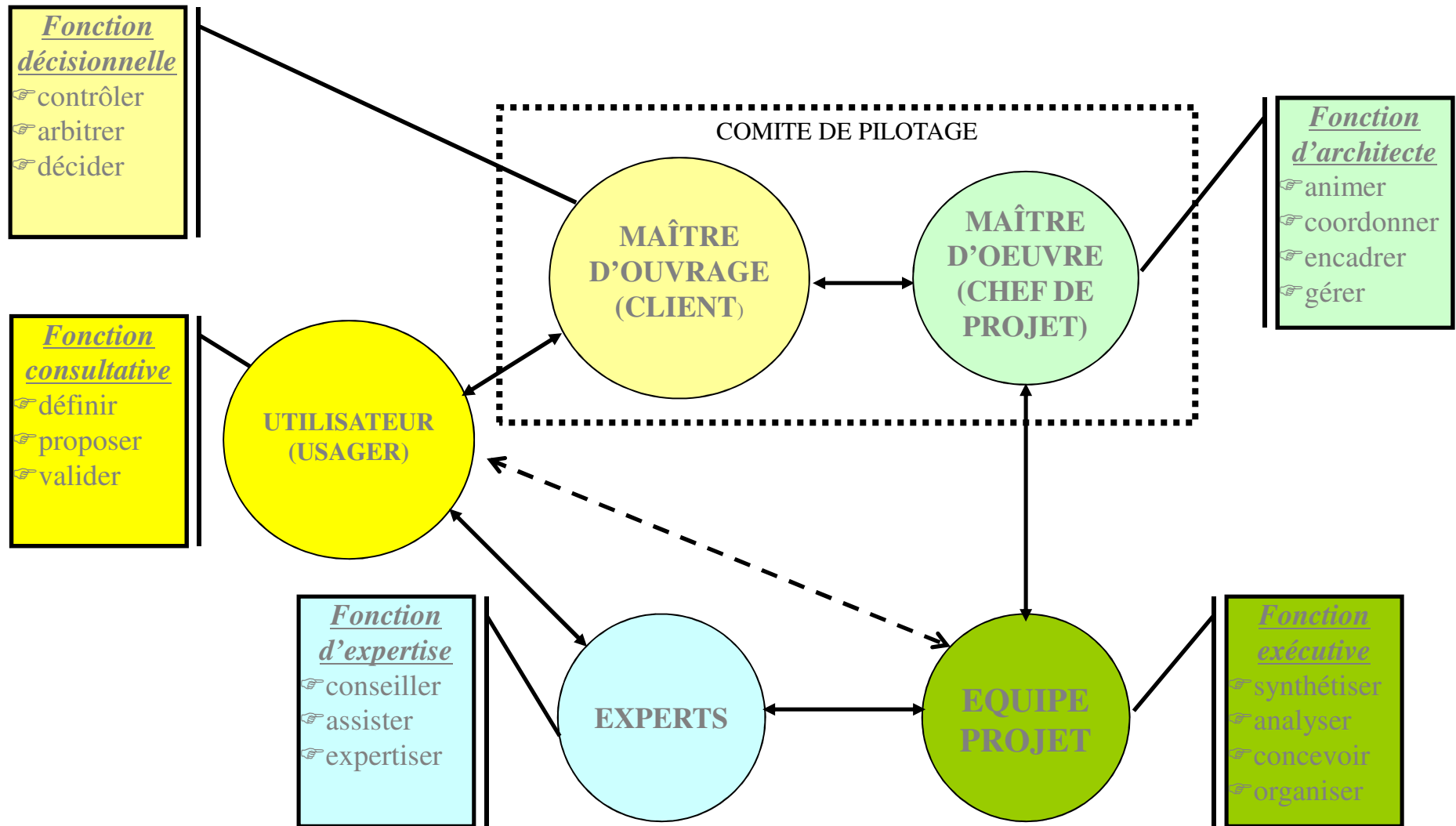
Le cycle de vie

*L'approche
Processus*



La problématique : les ingrédients

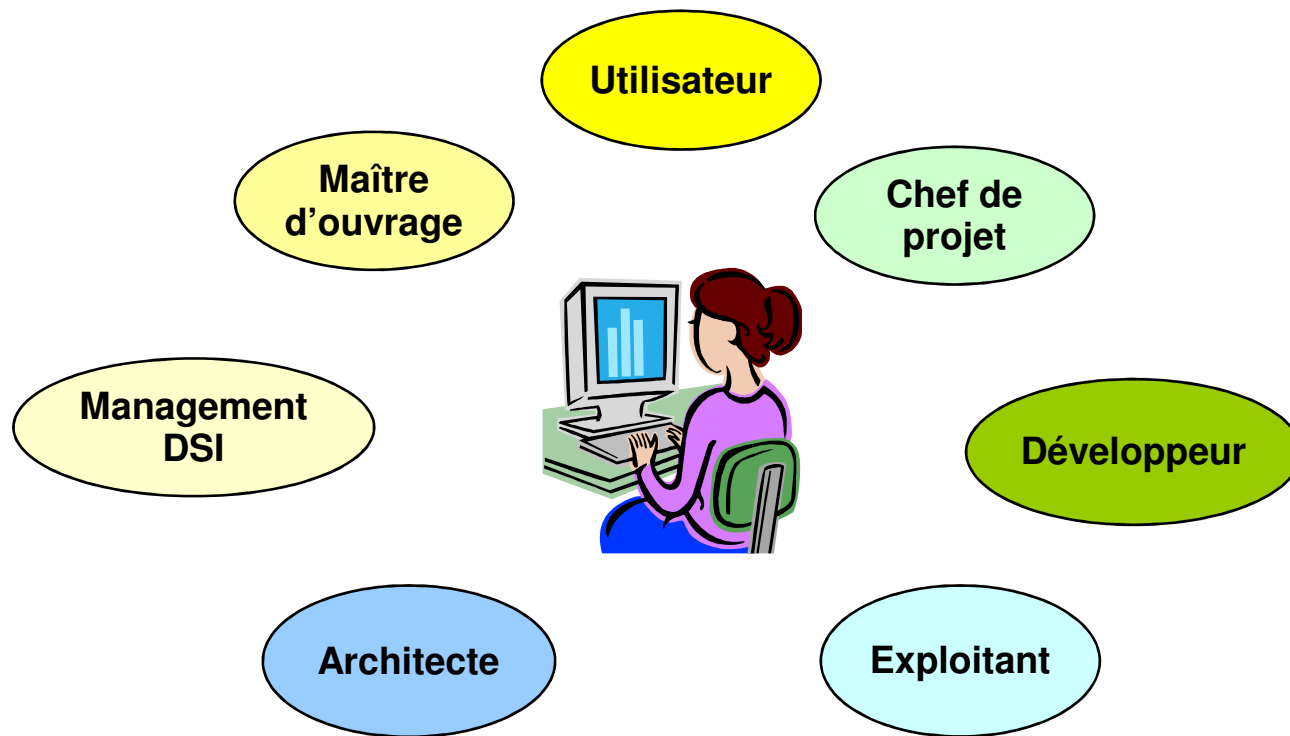
Les acteurs



La problématique : les ingrédients

Les acteurs

- Différents points de vue : qu'est-ce qu'un logiciel de Qualité ?



La problématique : les ingrédients

Les acteurs

- Différents points de vue : qu'est-ce qu'un logiciel de Qualité ?

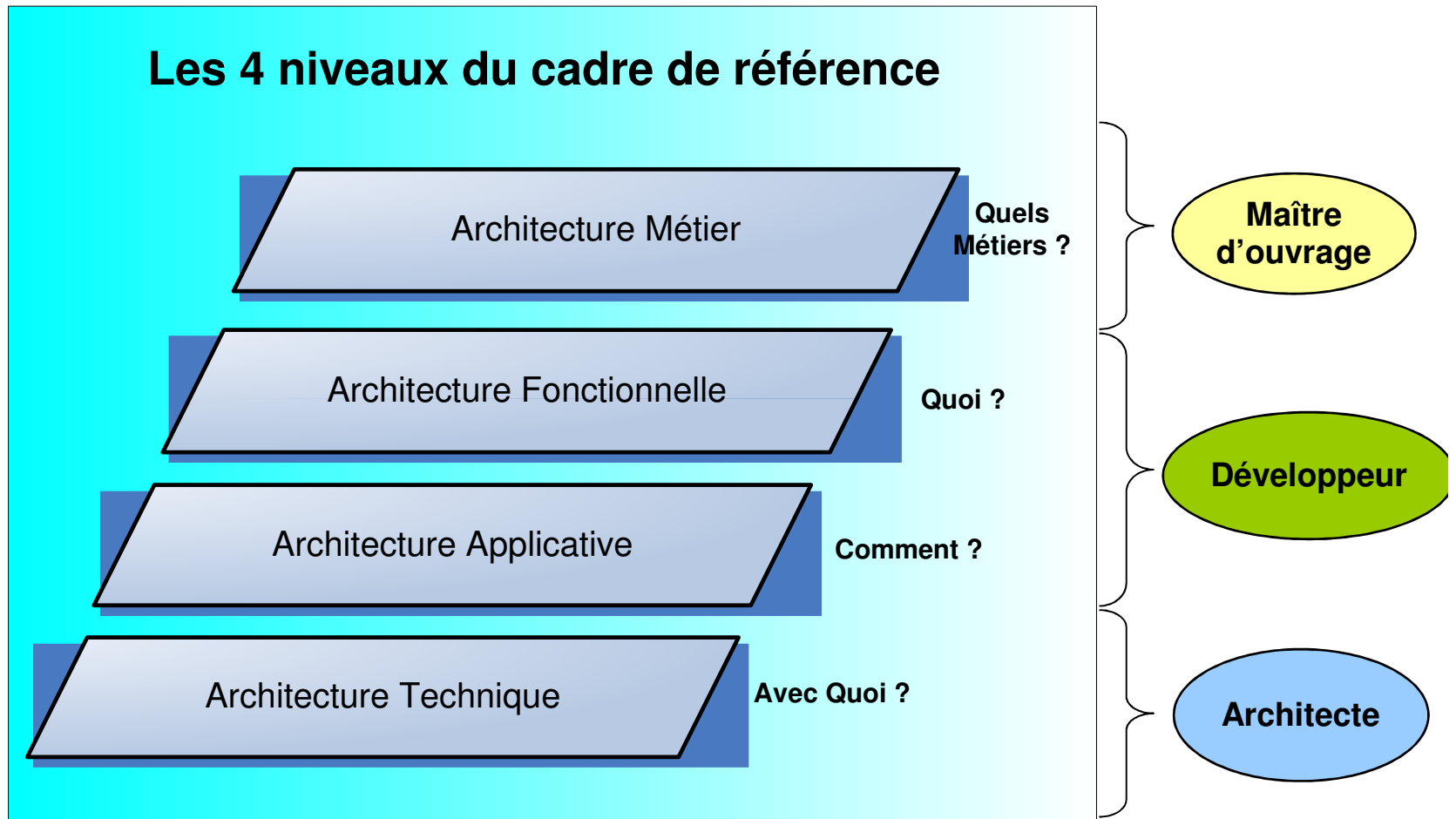


Mode Projet
<ul style="list-style-type: none">• <i>Faire bien du premier coup</i>• <i>Développer selon les règles de l'art</i>• <i>Prendre en compte les exigences MOA</i>• ...

Mode maintenance du patrimoine
<ul style="list-style-type: none">• <i>Fournir des éléments opérationnels pour identifier, évaluer et prioriser les différentes maintenances</i><ul style="list-style-type: none">✓ <i>Curative</i>✓ <i>Corrective</i>✓ <i>Evolutive</i>✓ <i>Adaptative</i>✓ <i>Préventive</i>

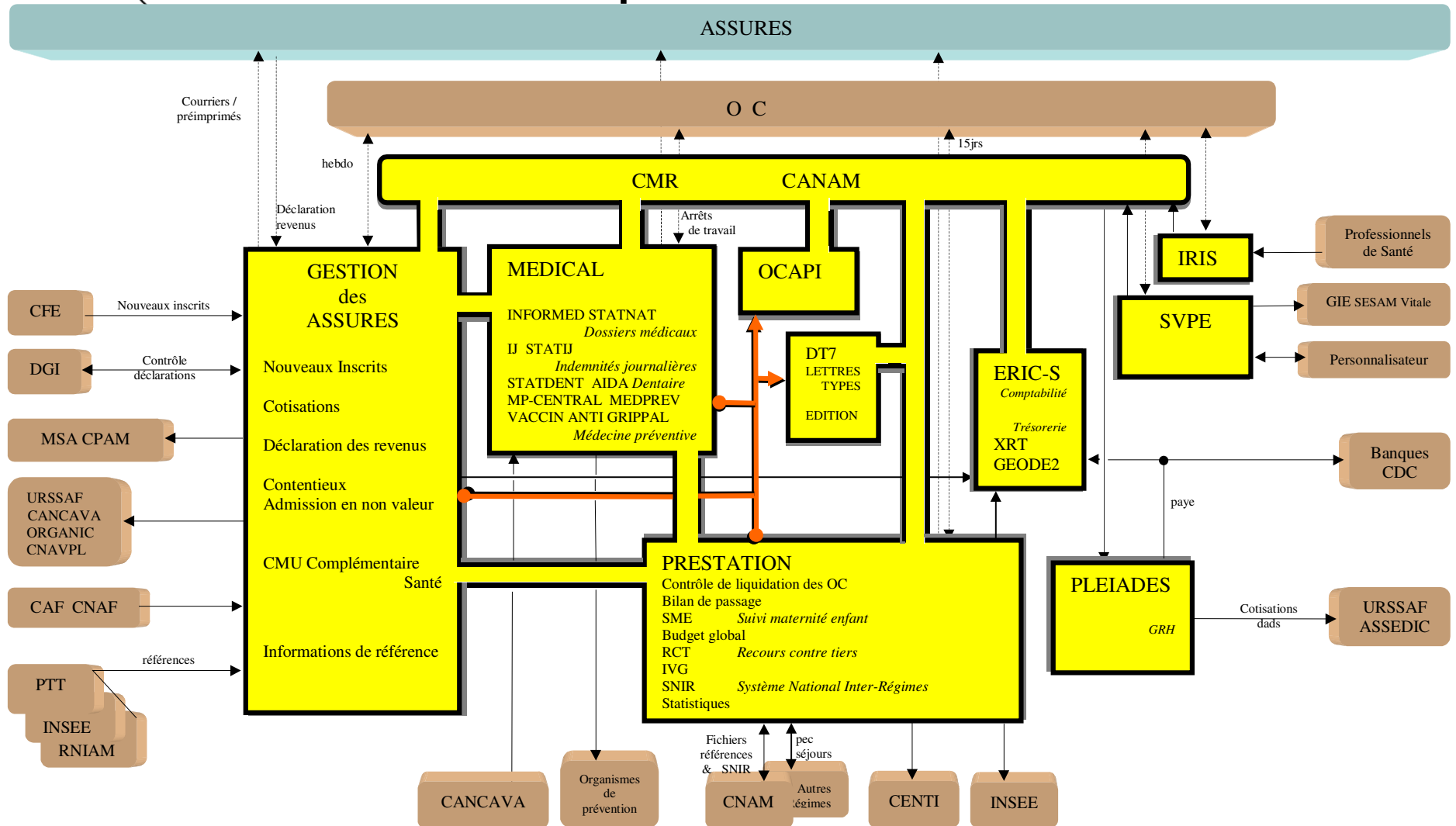
La problématique : les ingrédients

Les niveaux



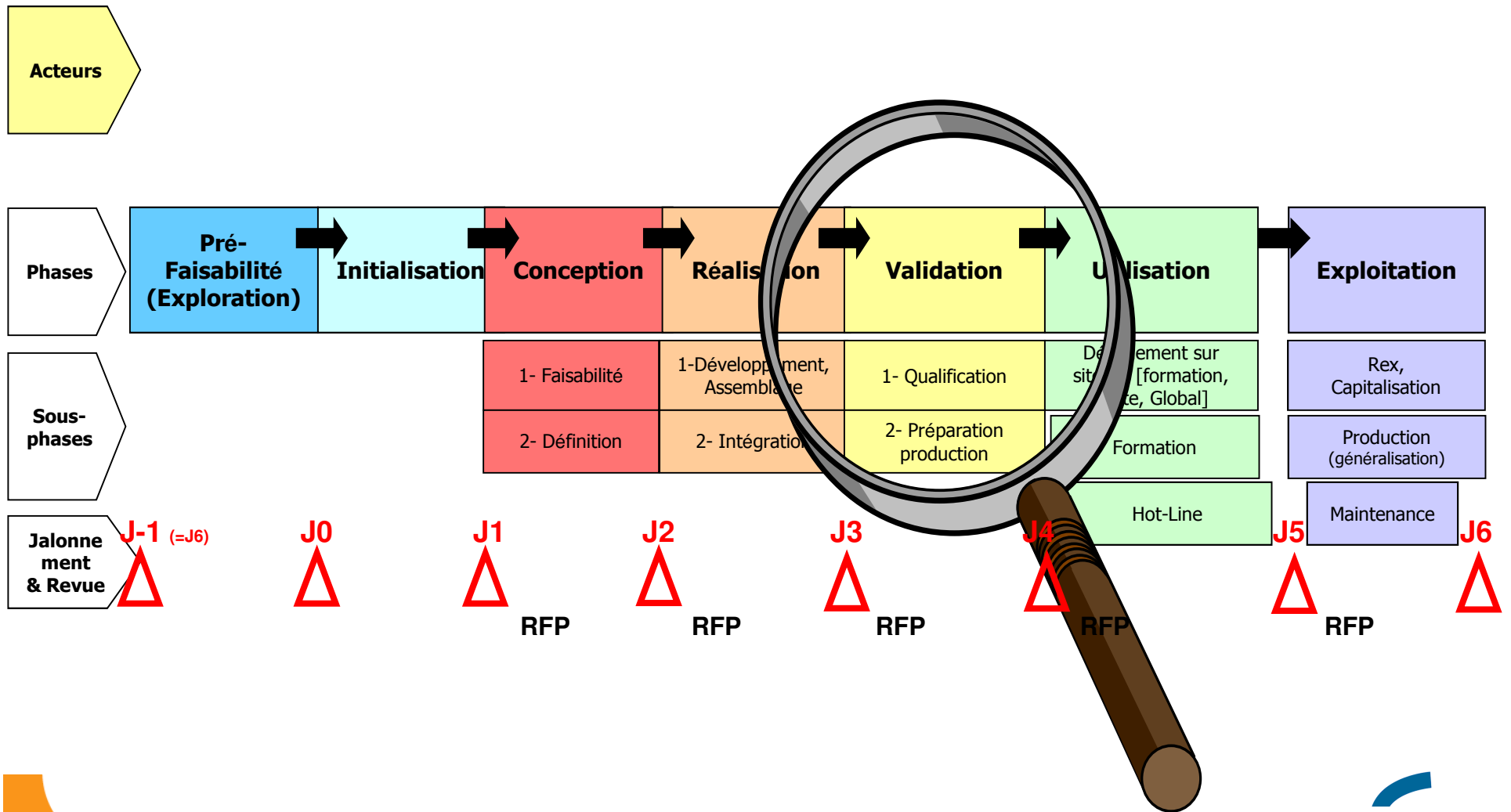
La problématique : les ingrédients

Les niveaux : un exemple d'architecture fonctionnelle



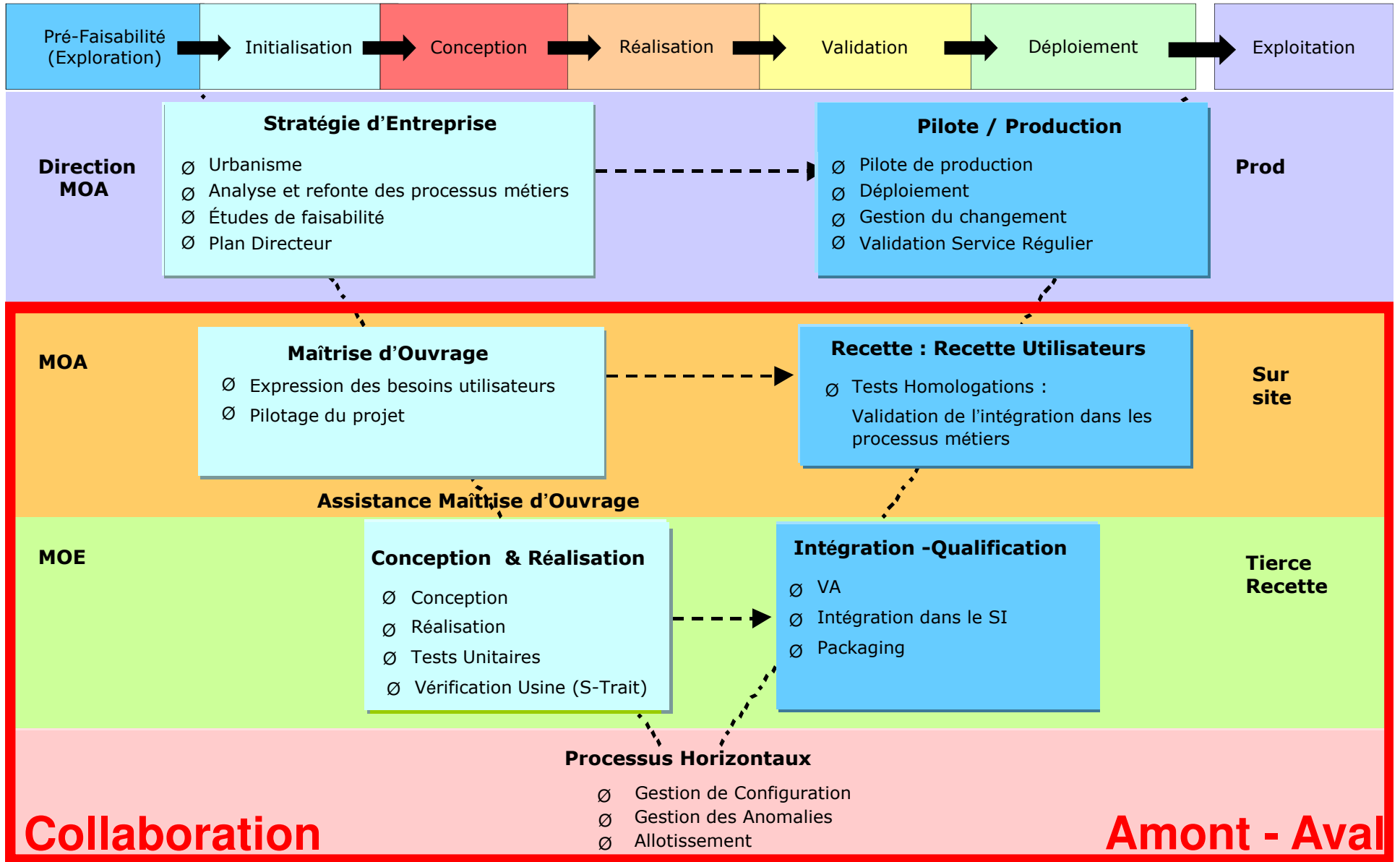
La problématique : les ingrédients

Le cycle de vie



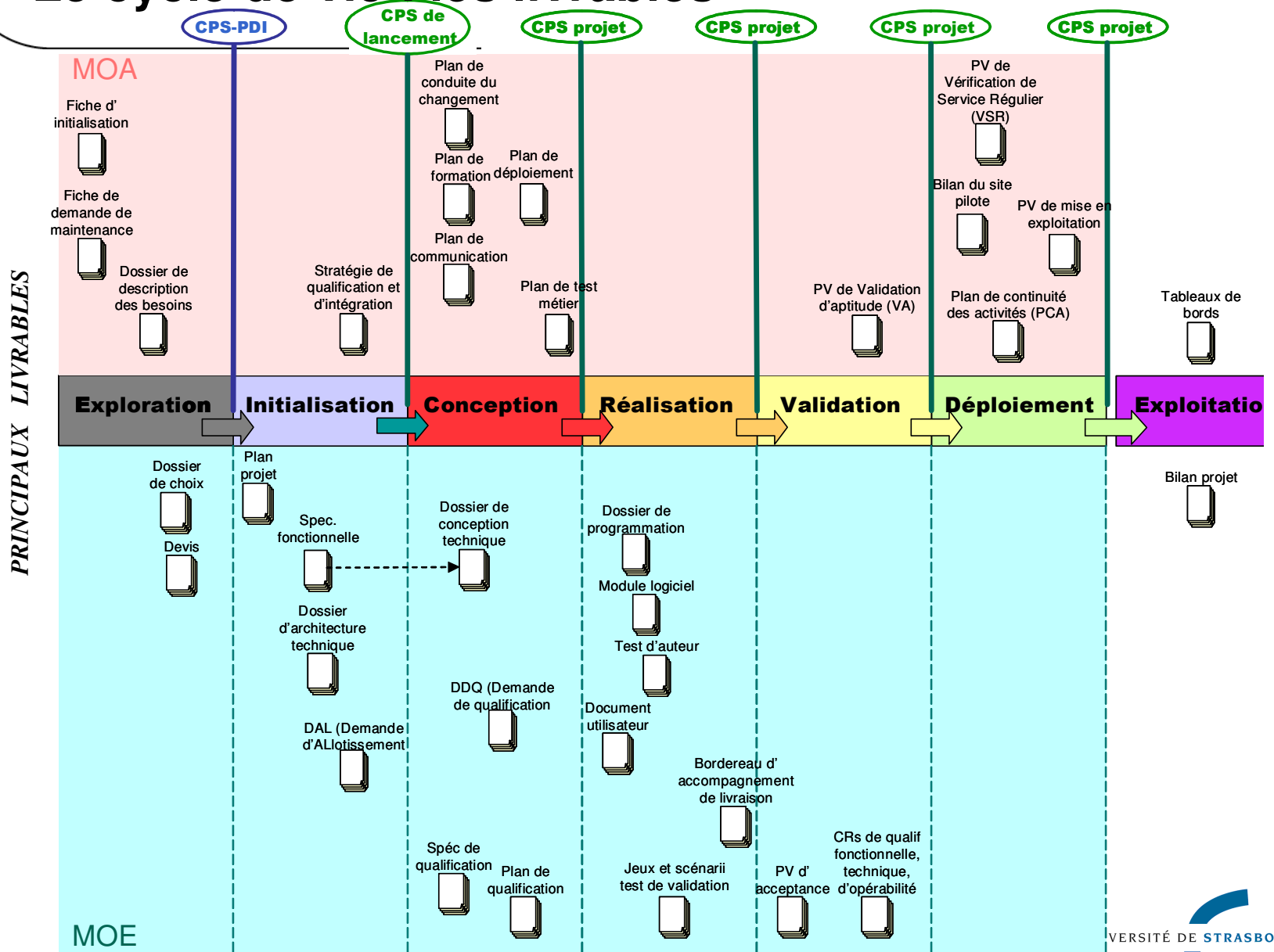
La problématique : les ingrédients

Le cycle de vie : les acteurs



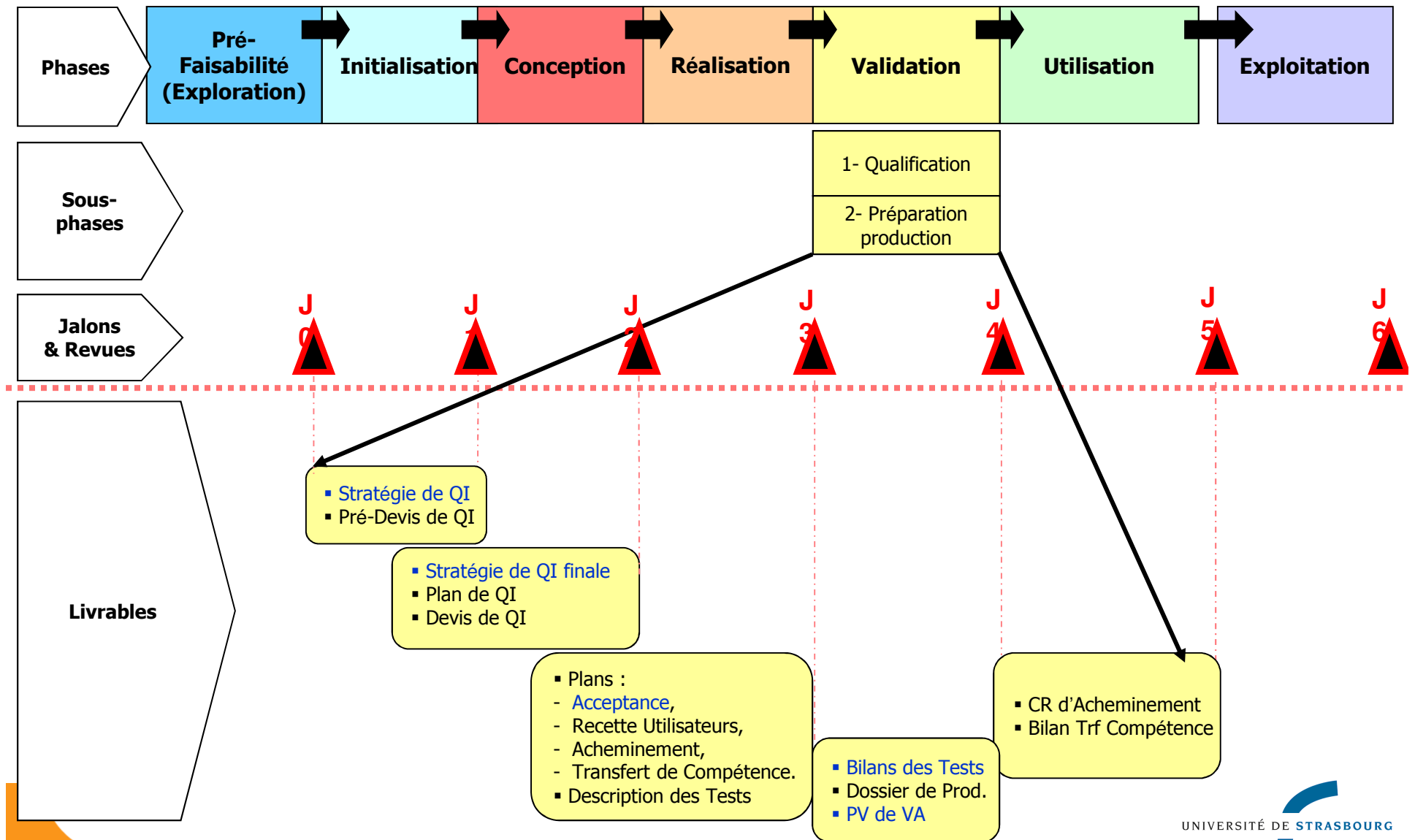
La problématique : les ingrédients

Le cycle de vie : les livrables



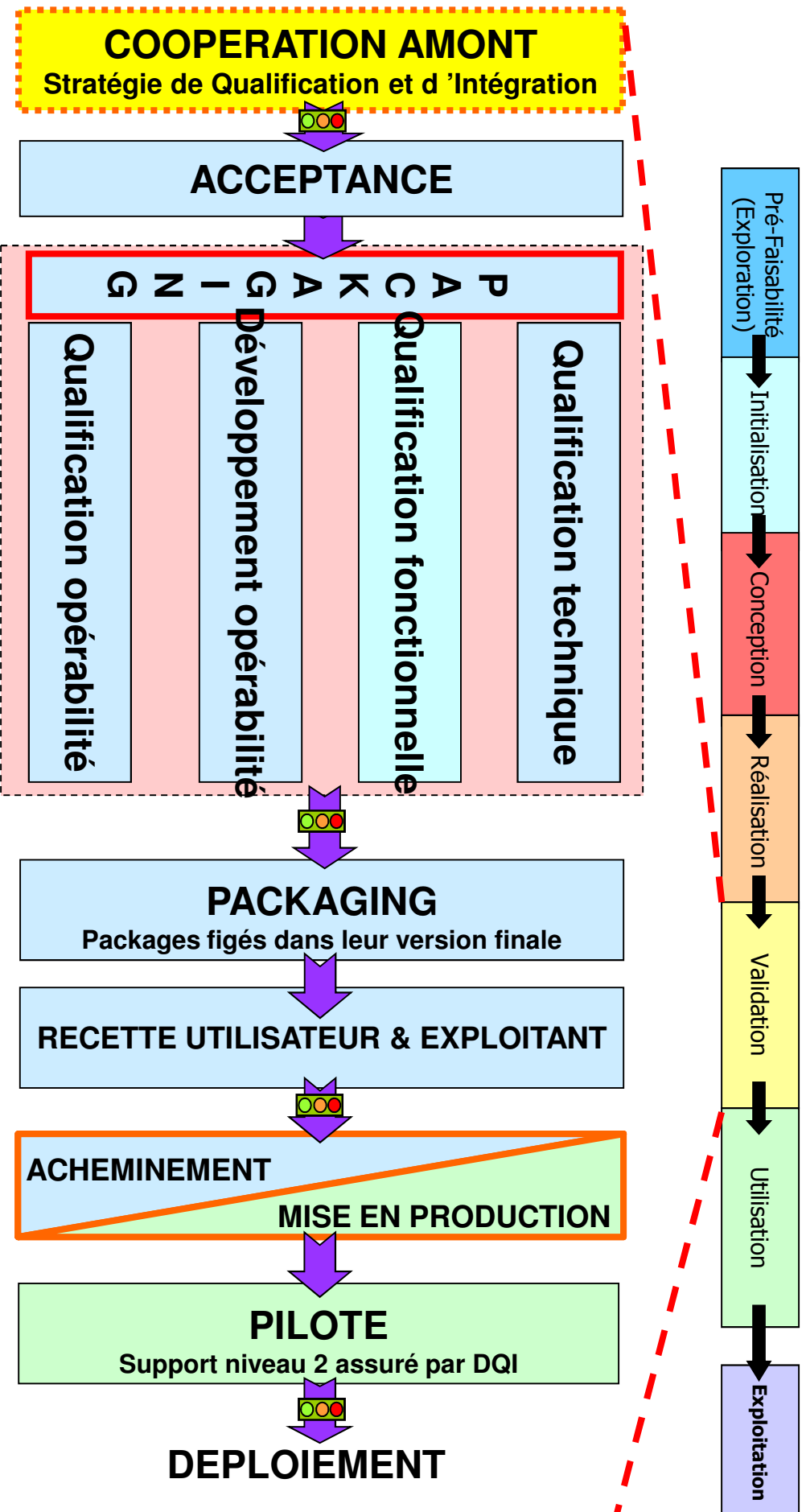
La problématique : les ingrédients

Le cycle de vie : les livrables



La problématique : les ingrédients

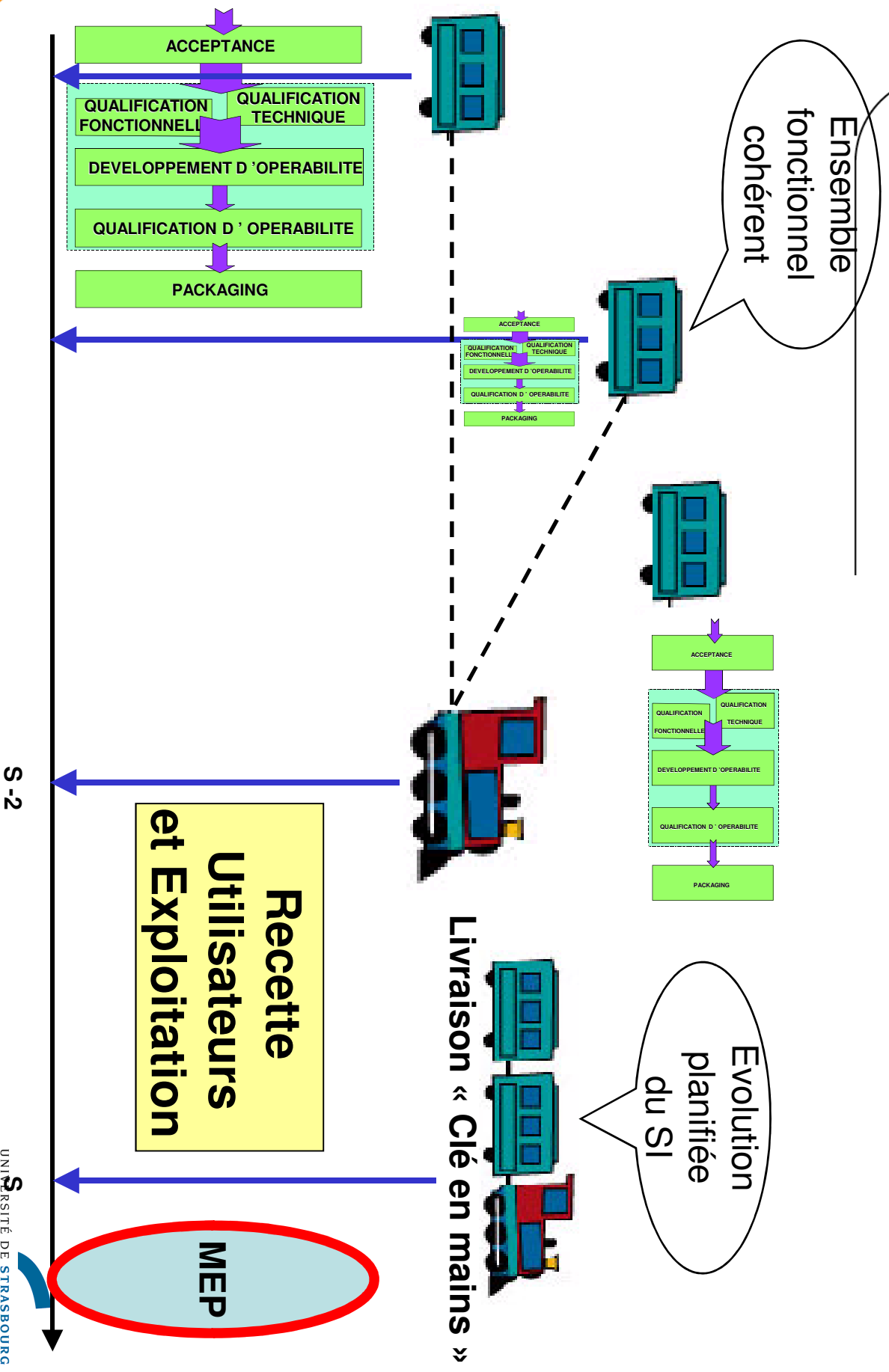
L'approche processus



Alloissement, Gestion de configuration, Gestion des anomalies

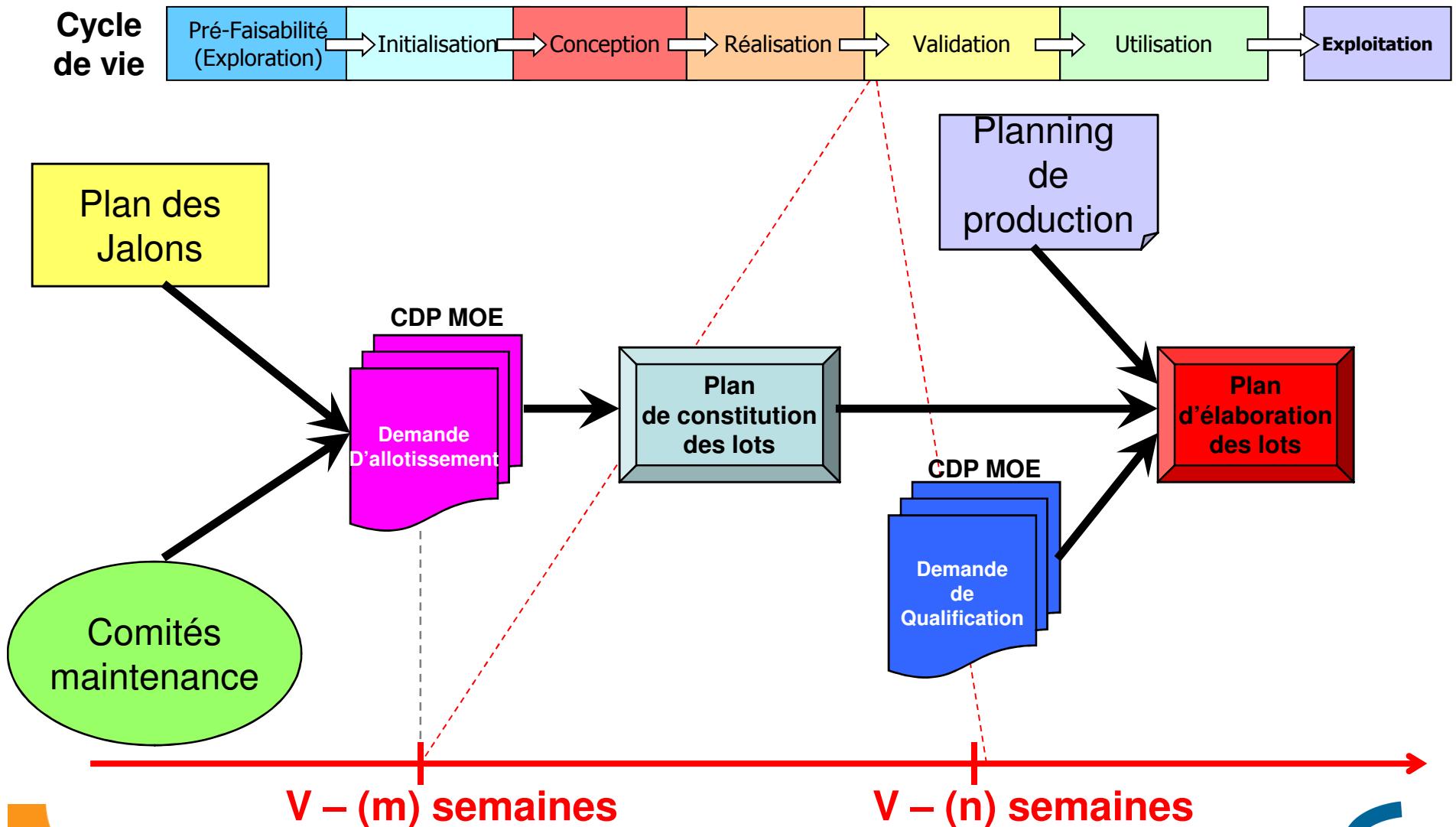
La problématique : les ingrédients

Le cycle de vie : l'industrialisation de la mise en production



La problématique : les ingrédients

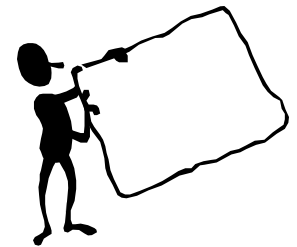
Le cycle de vie : la planification



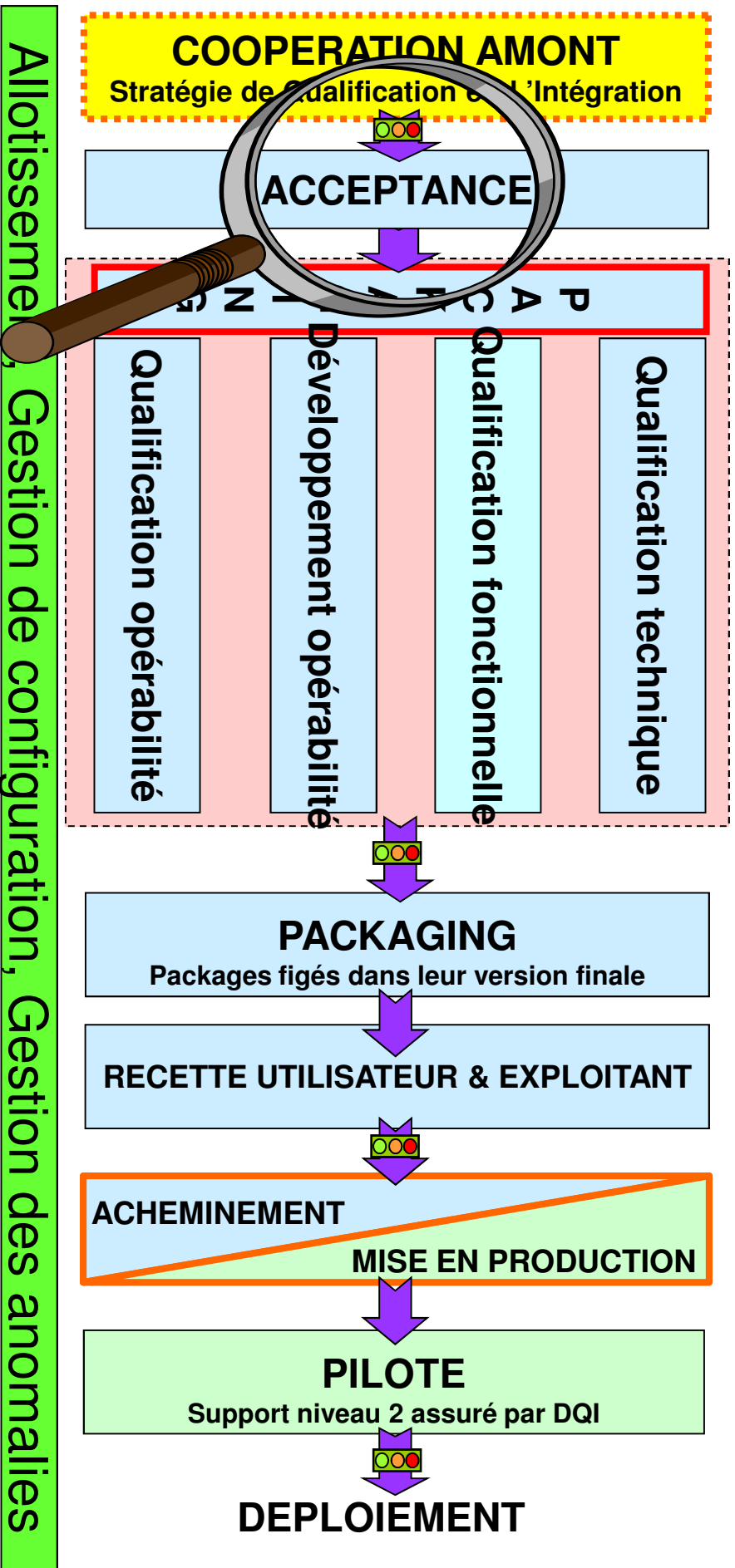
Plan de l'Intervention



2. Les processus d'intégration et de qualification

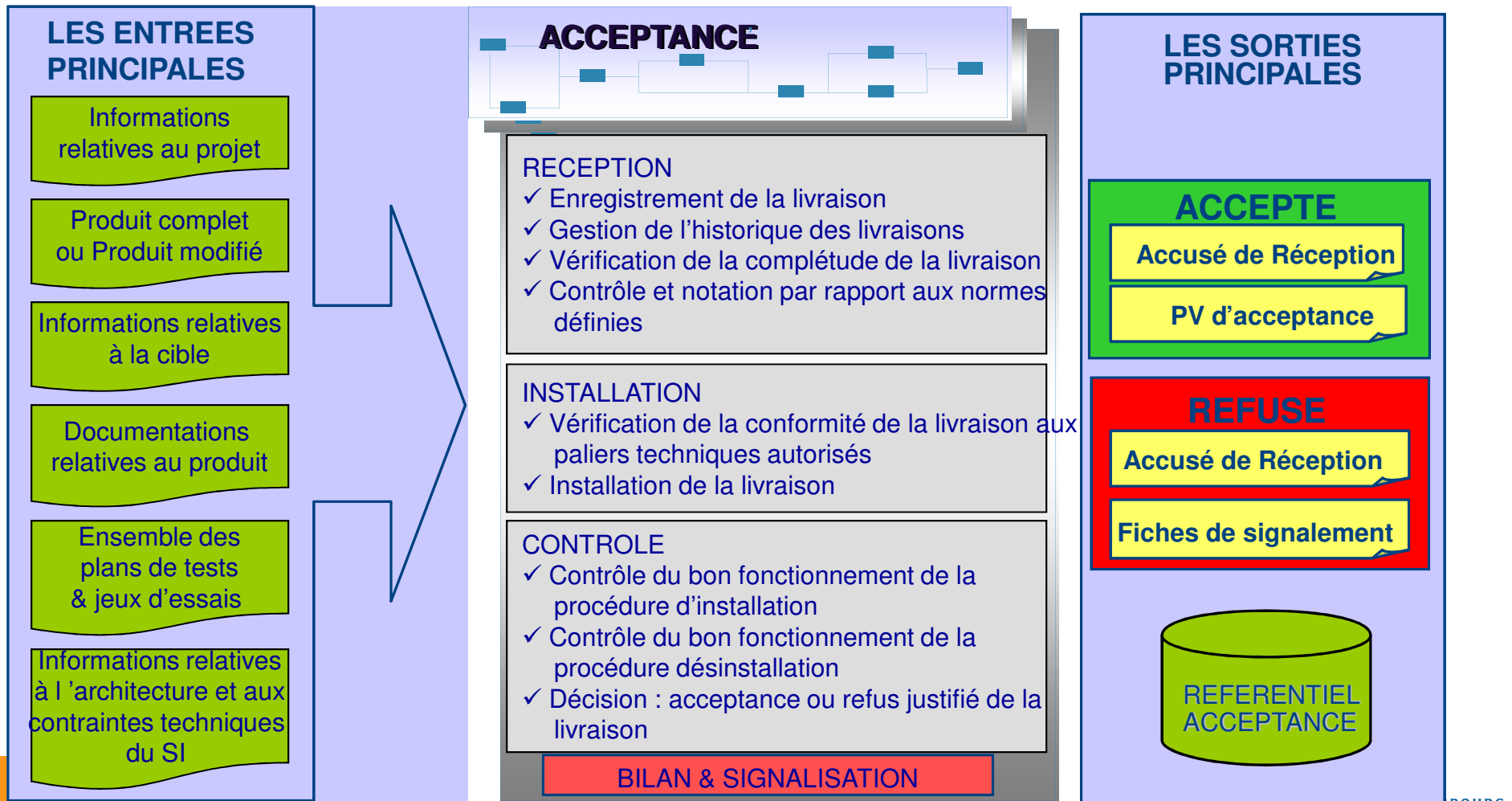


Les Processus de Qualification Intégration



Le processus d'Acceptance

Objectif : Vérifier que le produit est complet, installable et utilisable.



Le processus de Qualification Fonctionnelle

Objectif : Vérifier la conformité du produit au cahier des charges sur le plan fonctionnel



Le processus de Qualification Fonctionnelle

Il s'agit de :

- Contrôler la conformité du produit par rapport à la demande spécifiée,
- Contrôler la conformité du produit (logiciel et documentation) par rapport aux normes métiers et aux standards utilisés,
- Contrôler la cohérence fonctionnelle du produit, tant interne (le produit considéré isolément) qu'externe (le produit et son environnement) avec étude de l'interaction fonctionnelle avec les autres éléments du SI.

Le processus de Qualification Fonctionnelle

Les utilisateurs MOA peuvent intervenir dans ce processus lors des sessions d'enregistrement des plans de tests permettant de vérifier :

- la conformité à la demande,
- la conformité aux normes métiers,
- la non régression fonctionnelle.

Le processus de Qualification Fonctionnelle

Contrôle de conformité par rapport à la demande spécifiée

La demande s'exprime le plus souvent par son cahier des charges (développement sous-traité) et ses spécifications fonctionnelles générales et détaillées (développement interne), mais d'autres expressions plus condensées sont possibles selon les types de demandes (actions curatives, correctives, préventives, ...).

Le logiciel doit être complet (couverture de la demande spécifiée), exact (le logiciel fait ce qui est spécifié) et stable pour les fonctionnalités non modifiées (test de non-régression).

Le processus de Qualification Fonctionnelle

Contrôle de conformité du produit aux normes de développement

La vérification de conformité du logiciel est faite par rapport aux normes de communication homme-machine (ergonomie des dialogues, présentation des éditions, ...)

La vérification de la documentation associée se fait par rapport aux normes documentaires (exhaustivité, présentation, contenu type, lisibilité).

Le processus de Qualification Fonctionnelle

Contrôle de cohérence fonctionnelle externe

Le contrôle s'assure de la cohérence fonctionnelle des interfaces entre le produit / version à qualifier et les produits environnants, ou les partenaires externes (clients, Etat, ...).

Ce contrôle avec l'environnement externe s'appuiera sur la mise en œuvre de la qualification fonctionnelle inter applications.

Pour les applications de taille importante, il est possible de procéder à une QF “ par briques cohérentes ” à condition qu'elles présentent un très faible niveau de dépendance.

La Qualification Fonctionnelle : problématique

Situation

- Définis (ou non) par la Maîtrise d'Ouvrage sans évaluation budgétaire
- Exécutés si le coût / délai le permet par la maîtrise d'œuvre
- Tâche a priori non valorisante
- De nombreux problèmes subsistent

Etat des Lieux

- Ce qui était supportable en terme de risques ne l'est plus
- Les coûts associés à la Non Qualité ressortent de plus en plus
- Les Démarches et Modèles ISO / CMMI / ITIL / ... en cours
- La sous-traitance (TMA, TRA) implique Recette et Contrôle

La problématique des Tests Fonctionnels

Comment établir la stratégie de tests ?

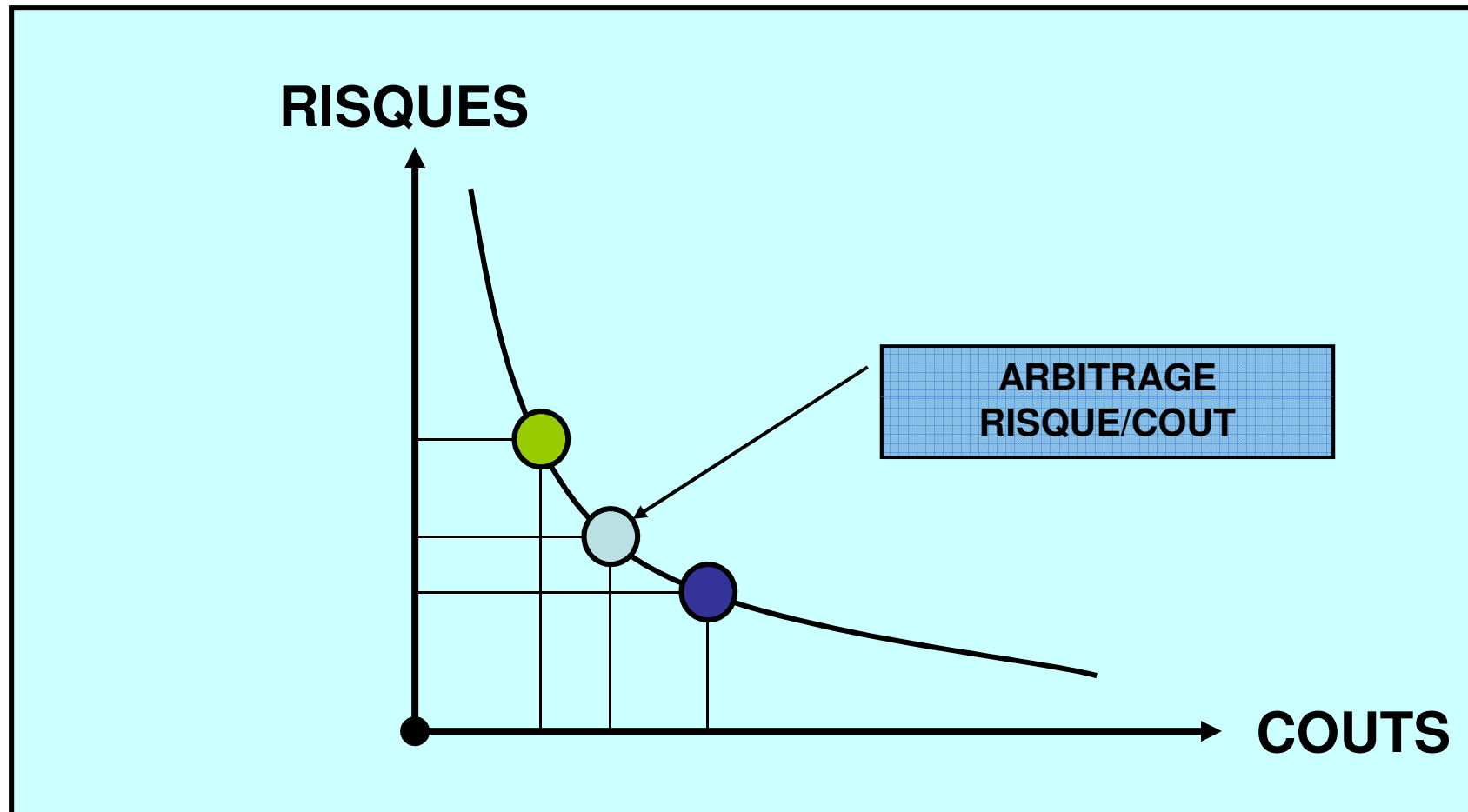
APPROCHE PAR

**La Maîtrise des
Risques**

DANS LE CADRE D'UN

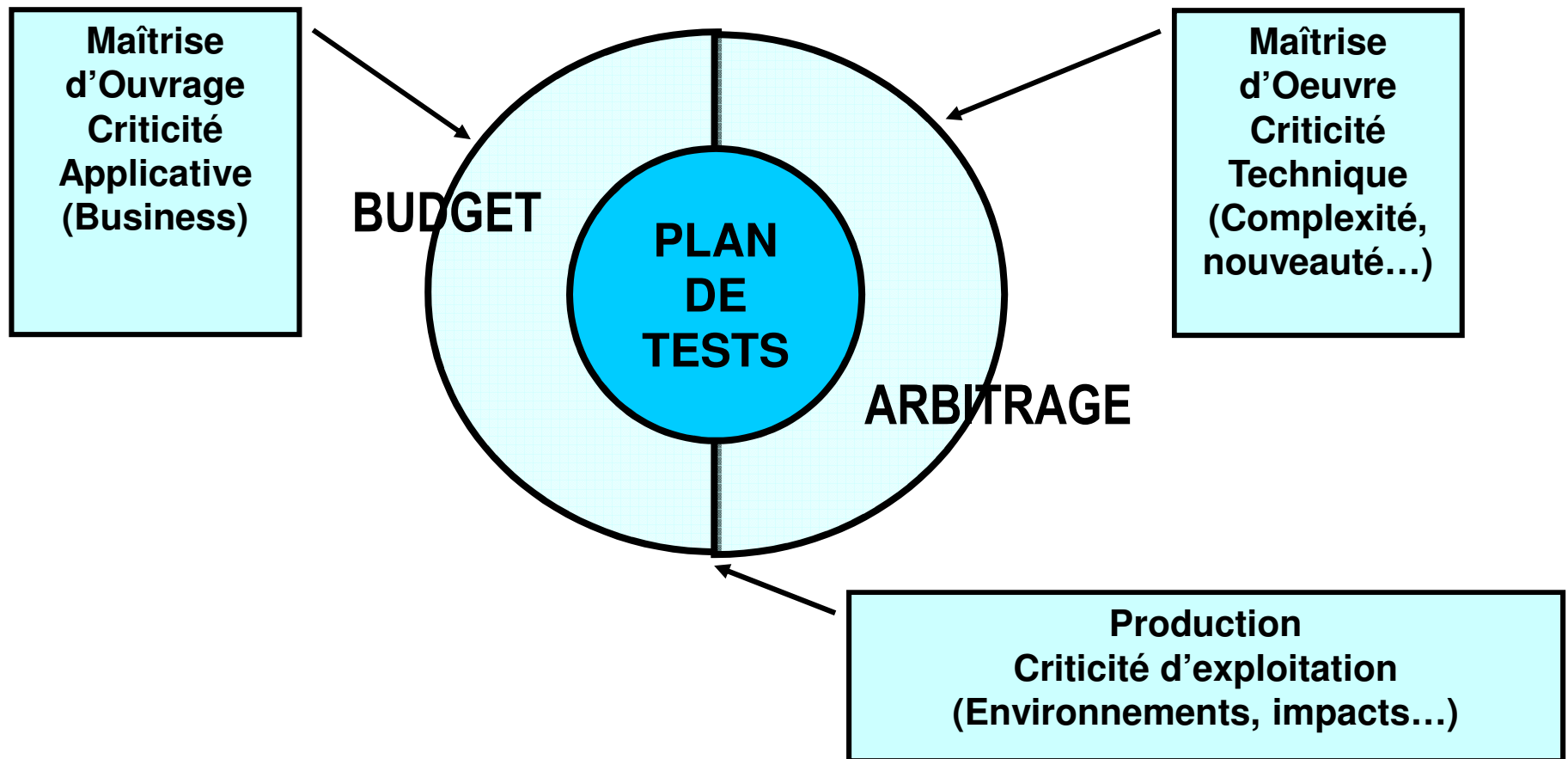
**Budget
défini**

La problématique des Tests Fonctionnels



La problématique des Tests Fonctionnels

Approche par les risques : Risk Based Testing (RBT)



La problématique des Tests Fonctionnels

Approche par les risques : Risk Based Testing (RBT)

Liste des fonctionnalités

Critères d'importance

Function / Requirement	Test Id 4 Char Code	Stabilité		Facteurs de Development						Special Factors			Scores and Indices			
		Homologation	Production	Impact sur la Release	Maturité fonctionnelle	Complexité de la Fonction	Tx Couverture des TU	Impacts Marketing	Impact dysfonctionne	Sécurité	Impacts sur Organisation		Beta Score	Integration Score	Beta Index	Integration Index
Pondération MOE		2	3	2	1	2	4	0	2	2	0	0				
Pondération MOA		0	4	0	3	4	0	5	5	4	4	0				
Connexion et habilitation																
Connexion portail fiscal	LAA1	1	2	1	1	1	1	1	5	5	5		37	85	3	4
Attribution du certificat	HBA1	1	1	1	1	1	2	1	5	5	5		38	81	3	3
Gestion Usager																
Création usager	CCL1	1	1	3	4	4	1	5	5	2	5		41	110	4	5
Modification usager	MCL1	1	2	1	2	2	1	3	3	2	3		30	72	2	3
Suppression usager	SCL1	1	2	4	3	4	1	1	1	1	2		35	55	2	2
Abonnement services	CAL1	0	0	4	1	5	1	2	4	2	3		35	73	2	3
Télédéclaration																
Simulation calcul IR	PCM1	1	3	4	1	5	1	5	5	3	5		50	117	5	5
Enregistrement déclaration	MCM1	2	2	3	3	3	3	3	4	3	3		51	88	5	4
Modification déclaration	ACM1	1	1	1	1	1	1	2	3	3	1		26	52	1	1
Visualiser reste dû	VCM1	3	3	3	1	3	3	5	3	2	5		50	95	5	4

Calcul automatique du nombre de cas de test « théorique »

	Beta	Integ
Total Requirements	80	80
Ranked 5	3	2
Ranked 4	3	4
Ranked 3	2	3
Ranked 2	4	3
Ranked 1	2	2
Ranked Requirements	14	14
Ignore = N/A for release	66	66

Estimated Tests	
Beta	Integ
90	60
60	80
30	45
40	30
10	10

Calcul automatique de la criticité

Priorité des processus

Importance pour l'utilisateur

	Stratégique	Important	Secondaire
Fréquence d'utilisation Forte	1	2	3
Moyenne	1	3	4
Faible	2	4	4

Valeurs fournies à titre d'indication : données modifiables



Unités d'Oeuvre

- Niveau de Complexité
- Contenu
- Pré Requis
- Profil Acteurs Nécessaires
- Coût d'une unité d'oeuvre

Détermination
d'Unités d'oeuvre

Industrialisation

OUTILLAGE DES TESTS

- Gestionnaire de Plans / Cas
- Générateur de données
- Automates de test

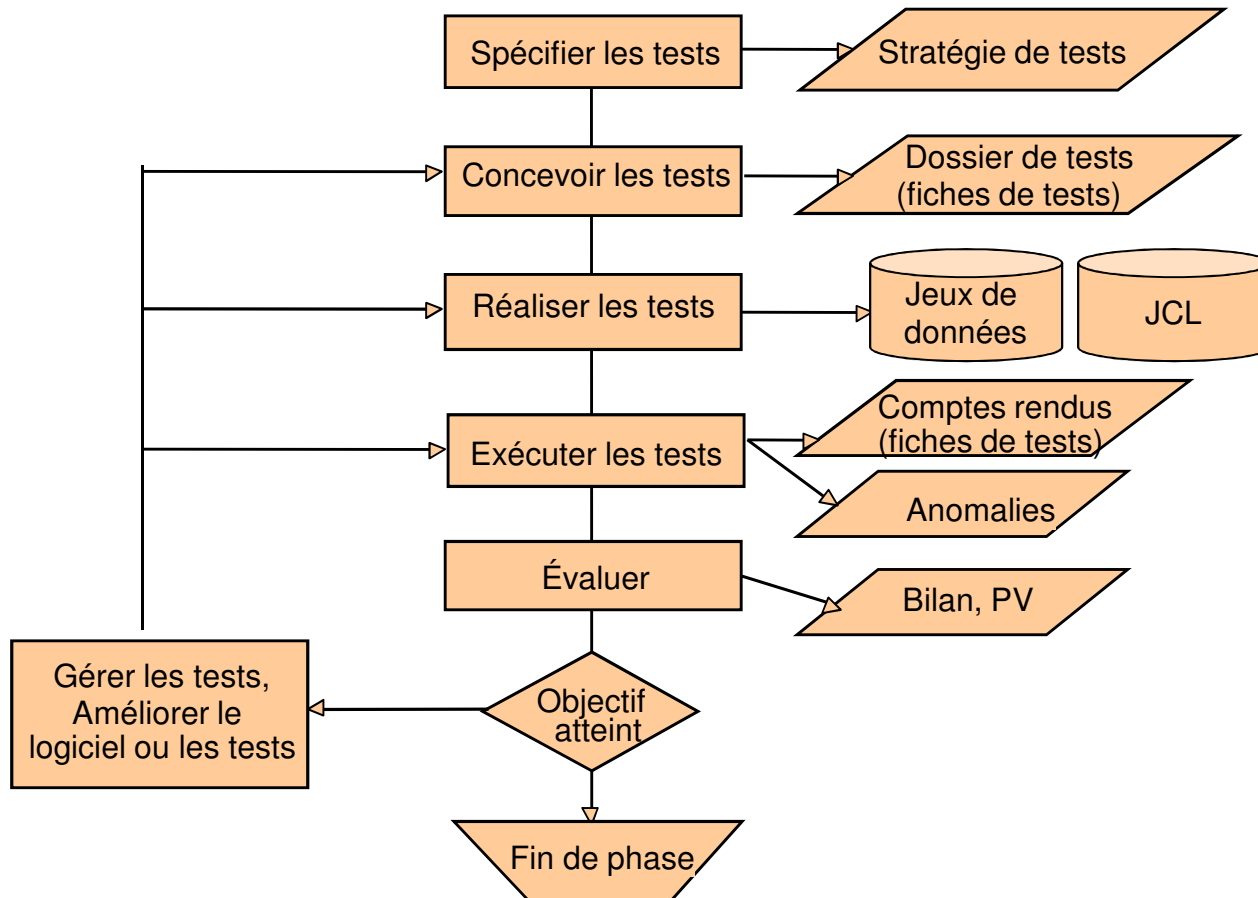
PROCESSUS

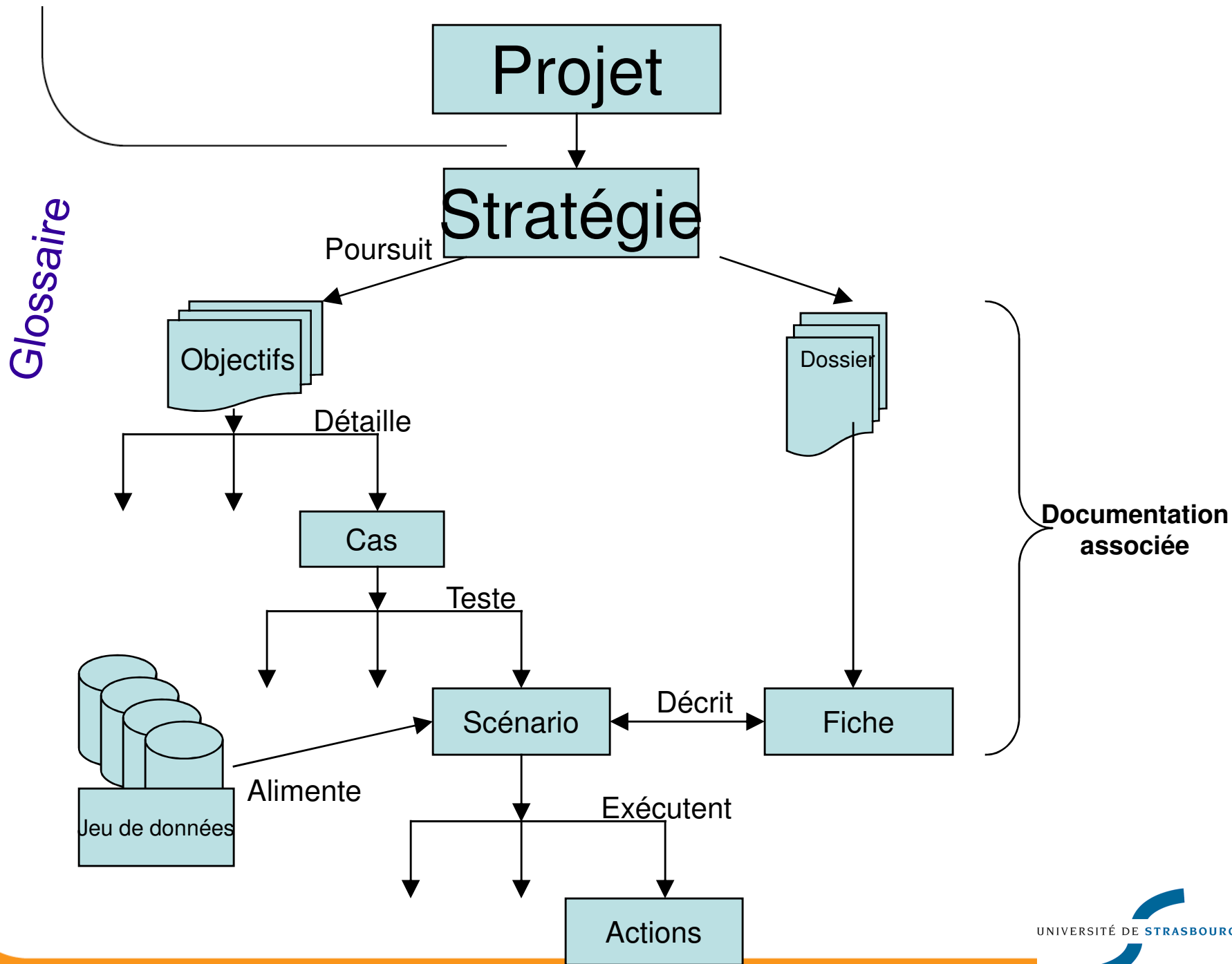
- Workflow
- Gestionnaire de Ressources

SUIVI TABLEAUX DE BORD

- Tableaux de bord
- Portail

Le cycle de vie du test





Quelques indicateurs

Processus

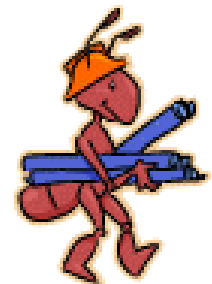
Mettre à disposition une application informatique

- Taux de mise en production dans les délais prévus
- Nombre d'incidents durant la première semaine de mise en production

Processus

Maintenir une application informatique

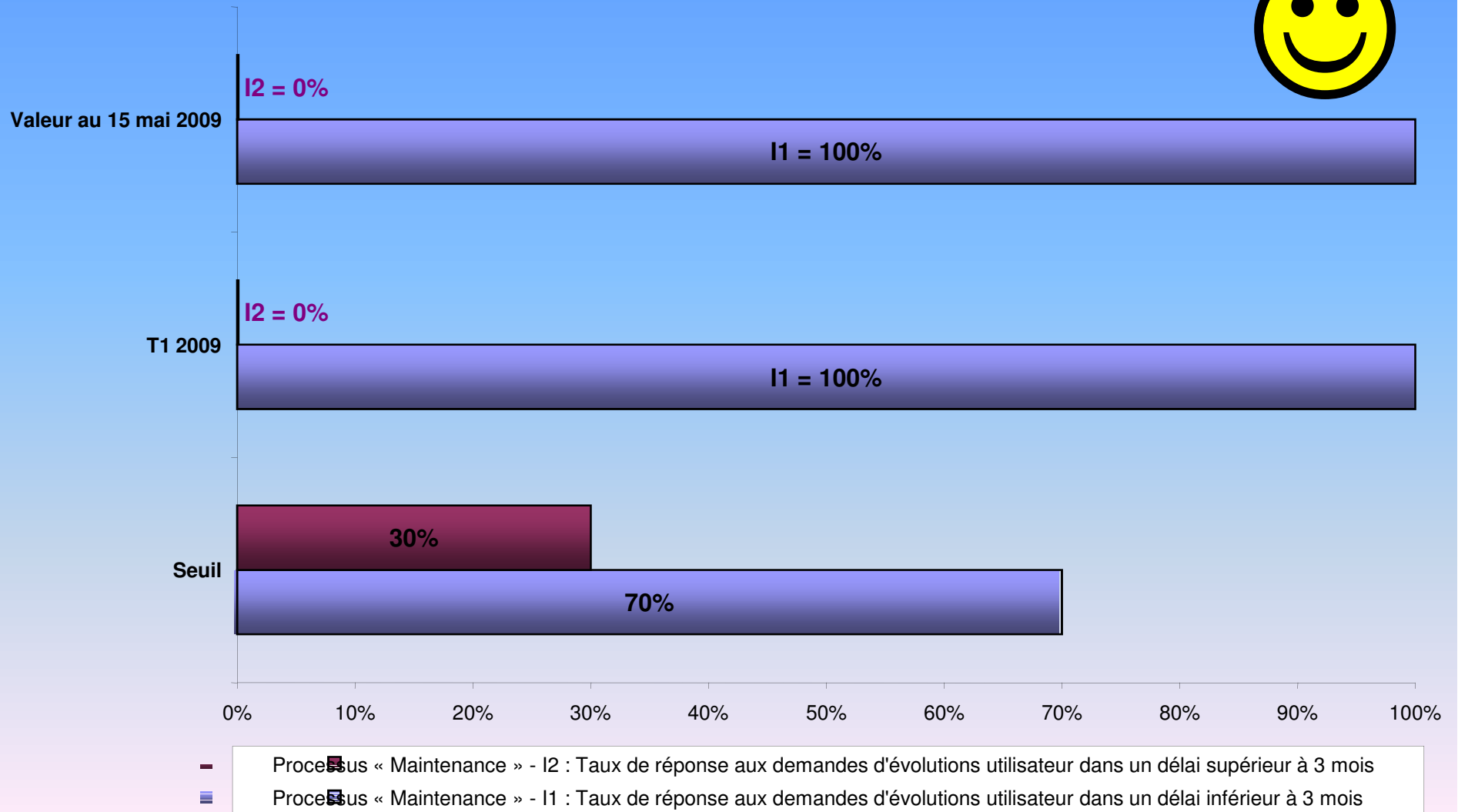
- Taux de réponse aux demandes d'évolution des utilisateurs



UNIVERSITÉ DE STRASBOURG

Portail

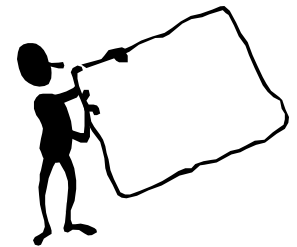
Taux de réponse aux demandes d'évolutions utilisateur



Plan de l'Intervention



3. Qualité du logiciel



UNIVERSITÉ DE STRASBOURG

Evaluation de la qualité logicielle

- La norme ISO 9126 : un référentiel pour évaluer la qualité d'un logiciel :

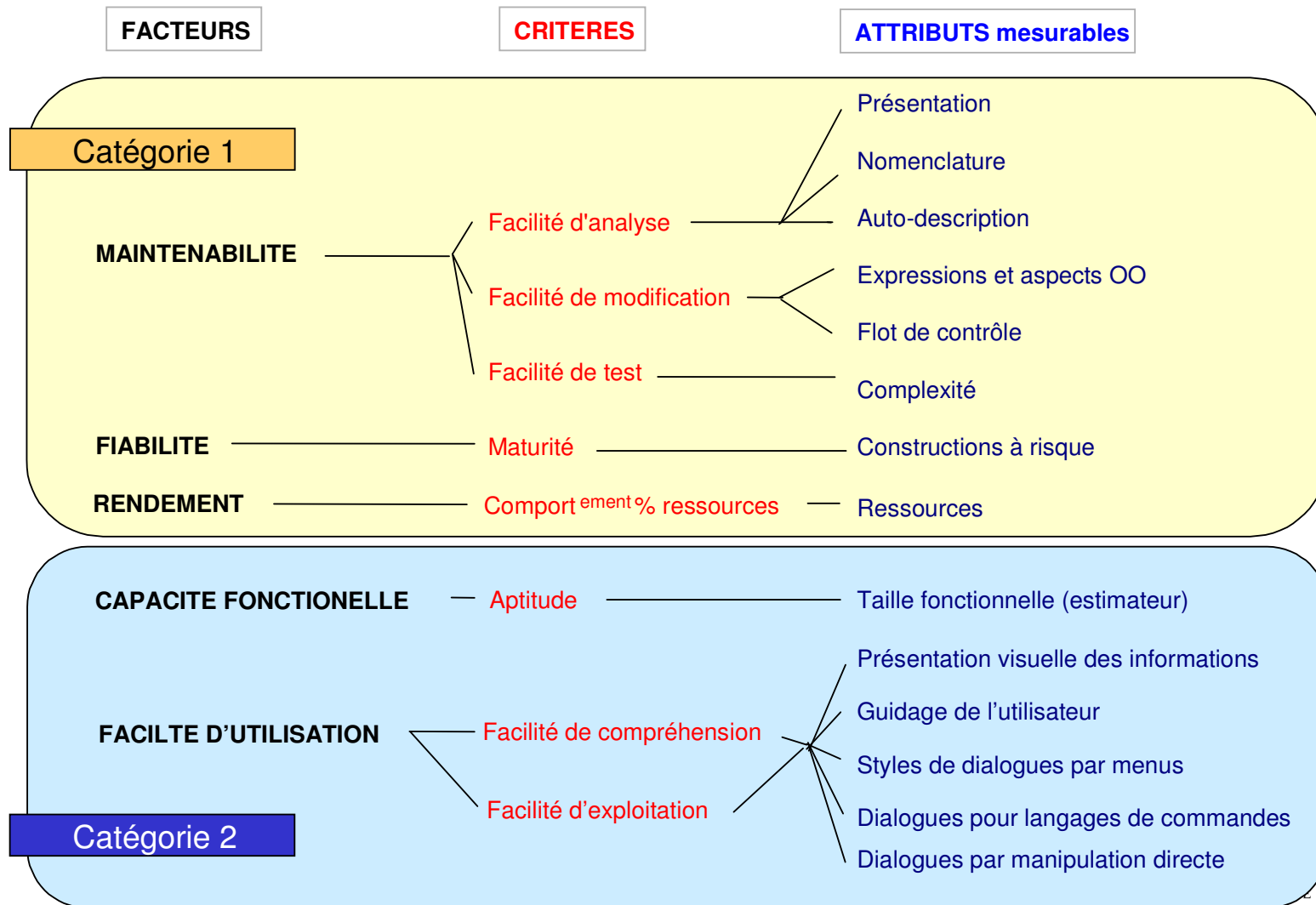
Facteur	Critères
Capacité fonctionnelle	<ul style="list-style-type: none">■ Aptitude■ Exactitude■ Interopérabilité■ Conformité réglementaire■ Sécurité
Fiabilité	<ul style="list-style-type: none">■ Maturité■ Tolérance aux fautes■ Possibilités de récupération

Evaluation de la qualité logicielle

Facteur	Critères
Facilité d'utilisation	<ul style="list-style-type: none">■ Facilité de compréhension■ Facilité d'apprentissage■ Facilité d'exploitation
Rendement	<ul style="list-style-type: none">■ Comportement vis-à-vis du temps■ Comportement vis-à-vis des ressources
Maintenabilité	<ul style="list-style-type: none">■ Facilité d'analyse■ Facilité de modification■ Stabilité■ Facilité de test
Portabilité	<ul style="list-style-type: none">■ Facilité d'adaptation■ Facilité à l'installation■ Conformité aux règles de portabilité■ Interchangeabilité

Evaluation de la qualité logicielle

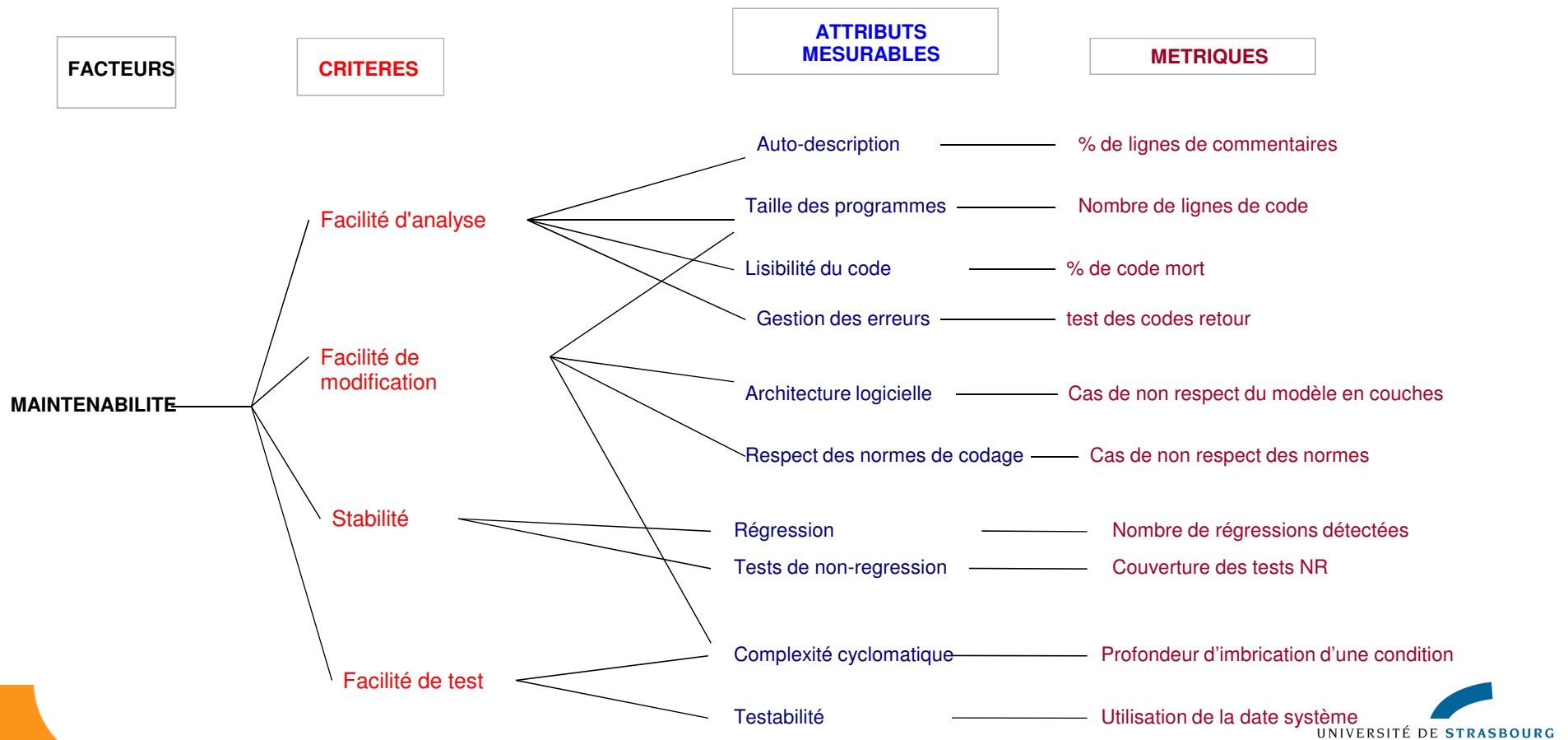
- Des attributs mesurables associés aux critères Qualité



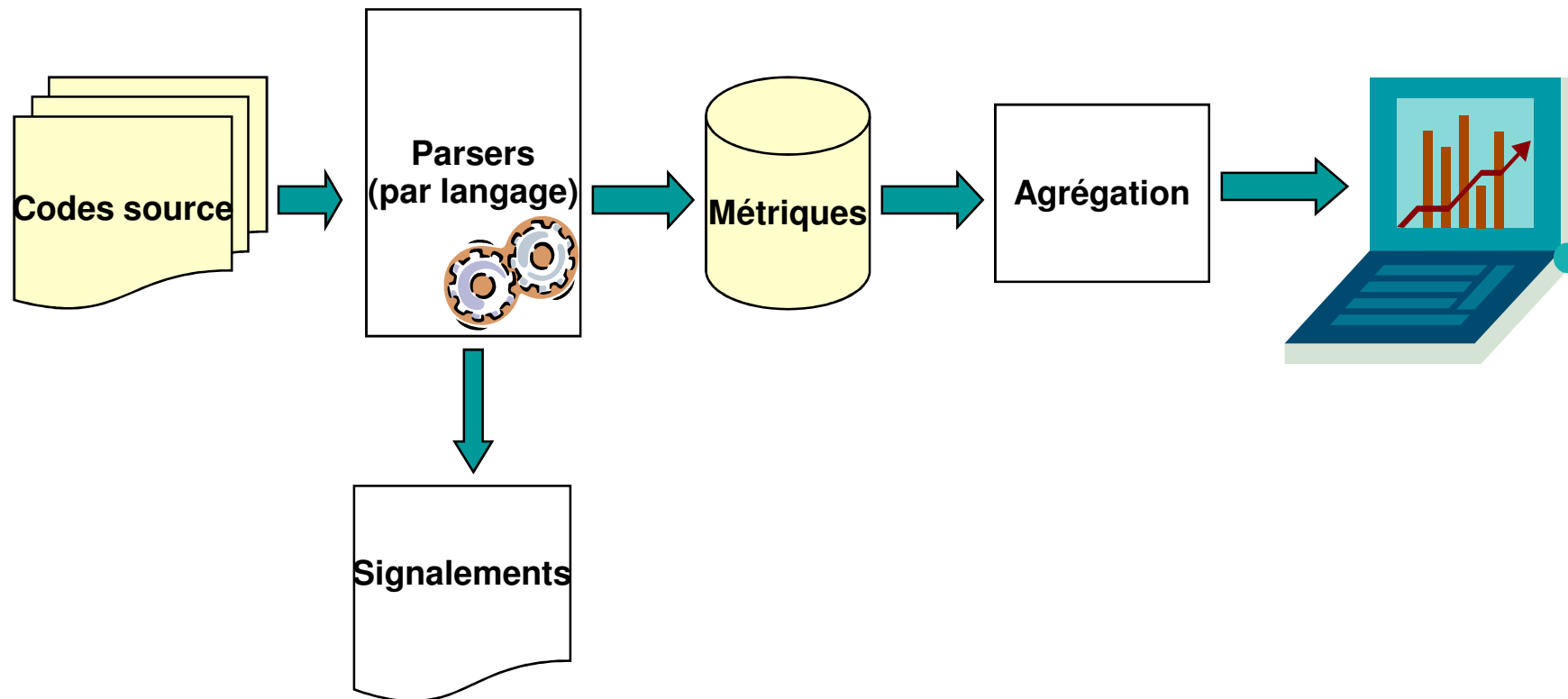
Evaluation de la qualité logicielle



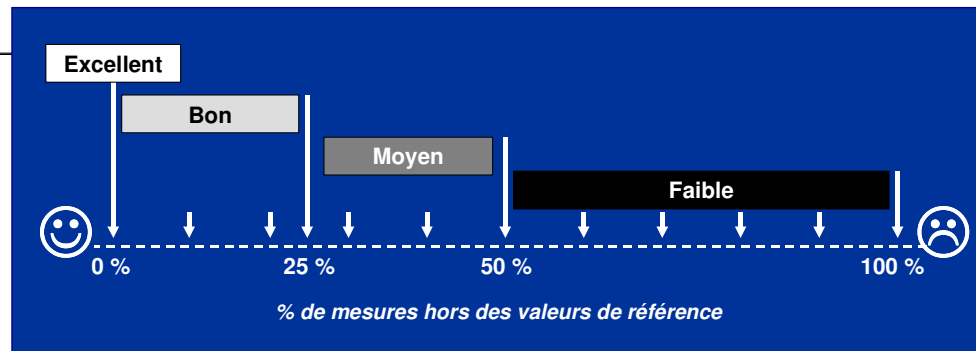
- Des métriques associés aux attributs mesurables :



L'analyse de code (qualimétrie)



L'analyse de code (qualimétrie)



MESURES ET SEUILS POUR LES APPLICATIONS DU TYPE « ANCIENNE TECHNOLOGIE »

Mesures	Seuil
Nombre de variables de premier niveau	30
Nombre de lignes d'instruction en procédure division	1000
Taux de commentaires en procédure division	0,2
Nombre cyclomatique (VG) en procédure division	250
Nombre cyclomatique (VG) moyen par paragraphe	4

MESURES ET SEUILS POUR LES APPLICATIONS DU TYPE « NOUVELLE TECHNOLOGIE »

Mesures	Seuil
Nombre d'attributs	15
Nombre cyclomatique (VG)	125
Taux de commentaire	0,2
Nombre de lignes de code	250
Nombre cyclomatique (VG) moyen par méthode	5

Cycle de vie

- La qualité sur les projets se construit tout au long du cycle de vie :
 - Dans les activités projet :
 - Gestion des exigences
 - Architecture
 - Développement
 - Gestion de configuration logicielle
 - Tests
 - ...
 - **Par une activité Assurance Qualité Logiciel :**
 - Référentiel des bonnes pratiques
 - Qualimétrie
 - Revues
 - Audits
 - Outillage d'auto-contrôle
 - ...

Assurance Qualité Logiciel

COMMENT ?

- **Les Analyseurs de Code Statique (standards Mc Cabe) produisent beaucoup de résultats**
- **L'Analyse Dynamique doit être prise en compte**
- **Le lien n'existe pas toujours entre analyse et correction**
- **Différences suivant les langages**
- **Freins psychologiques importants**
- **Accès natif aux outils complexes**

Assurance Qualité Logiciel

Simplifier l'accès aux analyses : outil maison





LIBRE SERVICE

- Portail d'accès
- Traitements rapides (1/2 J, J)
- Auto évaluation

SYNTHESE

- Modeles qualimetriques

Note

1		PAS ACCEPTABLE
2		INSUFFISANT
3		ACCEPTABLE
4		PARFAIT

Assurance Qualité Logiciel

QUAND ?

- **Processus continu en libre service pour les nouveaux développements**
- **Processus régulier en maintenance / TRA**
- **Processus renforcé à chaque « dérive »**
- **Etape obligatoire de la validation finale**

Assurance Qualité Logiciel

Enjeux

- Complément Tests Fonctionnels
- Diminution des coûts de tests
- Standardisation
- Contractualisation de la Qualité (sous traitance)
- Évaluation du Patrimoine
- Mesure du Progrès

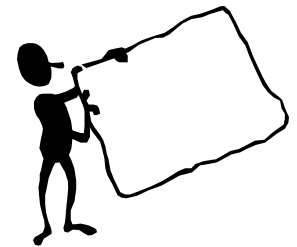
Priorités

- Modèle qualimétrique
- Accès en Libre Service
- Rapidité du Traitement
- A contractualiser

Plan de l'Intervention






4. Et plus loin encore



UNIVERSITÉ DE STRASBOURG

Passage du stade « artisanal »
au mode « Industriel »

Les 7 règles d'or

-  **1. La Communication**
-  **2. La Documentation : banque documentaire**
-  **3. La Collaboration « Amont »**



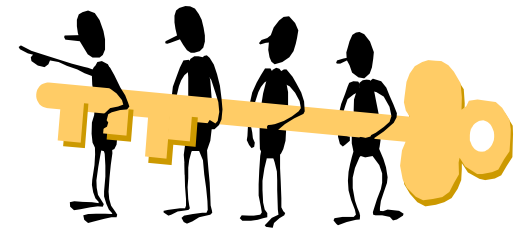
Les 7 règles d'or

4. La Formation

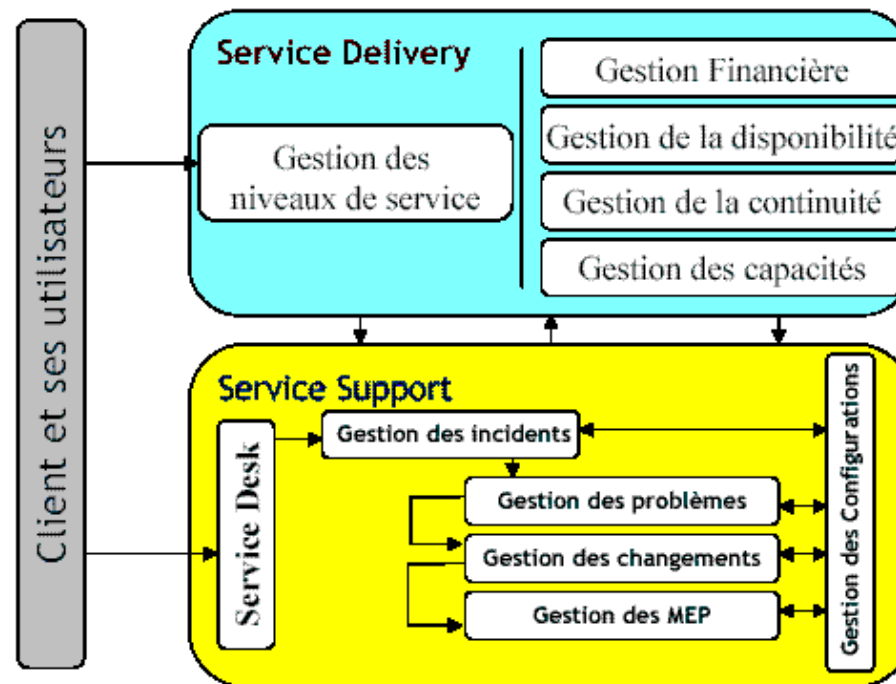
5. La Pratique

6. Le Poids de l'Existant

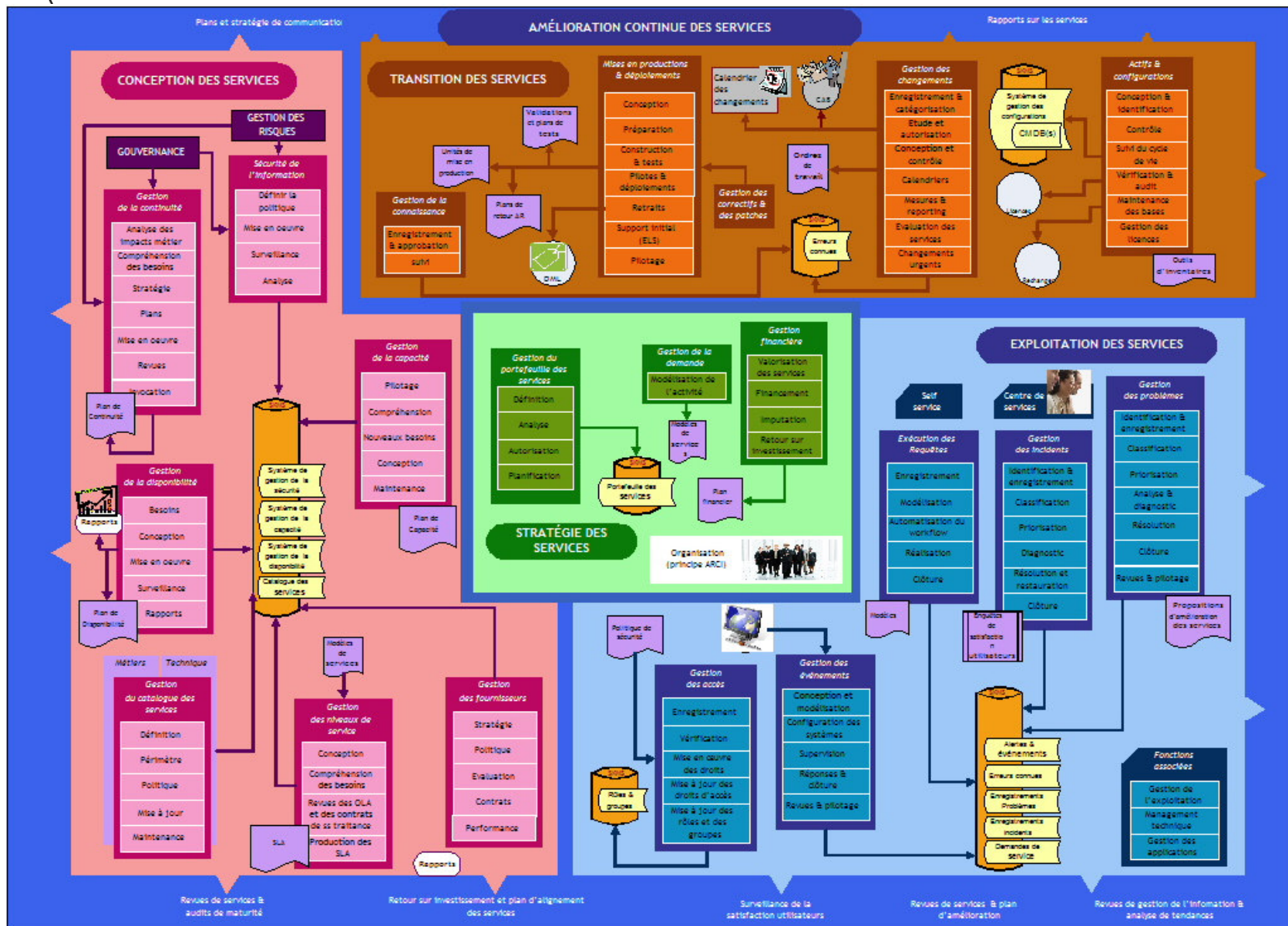
7. Les Processus Horizontaux



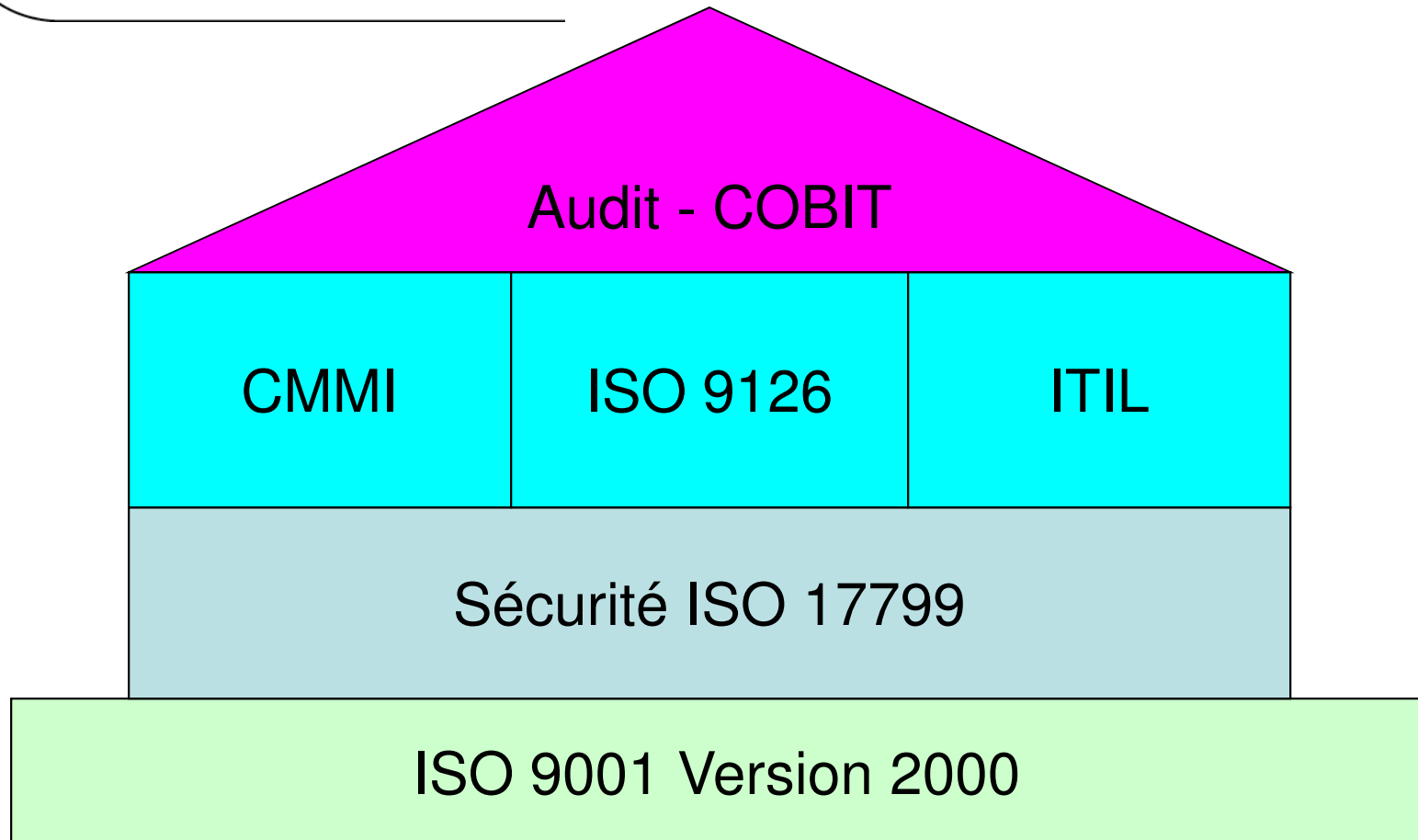
Processus ITIL de la gestion de services



Les référentiels et bonnes pratiques - ITIL V3



La maison des référentiels et bonnes pratiques



Merci de votre attention

