



Architecture des Systèmes d'Information

01 - Introduction

- ◆ **Le contenu de ce cours a été construit avec l'aide de :**
 - ▶ **Architecture Logicielle, Jacques Prinz, Dunod**
 - ▶ **Le Projet d'Urbanisation du SI, Christophe Longépé, Dunod**
 - ▶ **Mastering Archimate edition II, Gerben Weirda**
 - ▶ **Différents supports de cours nfe107 Urbanisation et Architecture des SI du CNAM,**

Plan du chapitre

1 Une infinité de cas...

2 Définitions / Les composants d'architecture SI

3 Démarche d'architecture SI

◆ Situations : votre analyse ?

▶ Incident de production :

- La baie du datacentre a rebootée samedi matin,
- L'ensemble des applications étaient redémarrées à 11H lundi.

▶ Saturation réseau :

- La migration de la messagerie vers gMail des employés du site de Nantes à saturé l'accès Internet de l'ensemble de la région ouest.

▶ Gestion de projet :

- Après avoir dépensé 40 % du budget total du programme dans la construction du POC, le fournisseur principal annonce qu'il n'est pas en mesure de respecter la contrainte du 5x9.

◆ Incident de production : reboot d'une baie

▶ Contexte métier :

- Quelles applications sont hébergées sur le DC ?
- Qui utilise ces applications ?
- Pour faire quoi ?
- Peuvent-ils s'en passer pendant le w.e. et le lundi matin ?

▶ Contexte technique :

- Pourquoi la baie a-t'elle rebootée ?
- Pourquoi le reboot de la baie a-t'il eu un impact sur les applications ?

▶ Contexte organisationnel :

- Le reboot a-t'il été détecté ?
- Les utilisateurs impactés ont-ils été prévenus ?

◆ Saturation réseau : migration à gMail

▶ Contexte métier

- Qui utilise Internet dans la région ouest ?
- Pour quelle activité ?

▶ Contexte technique

- Était-ce prévu ?
- Lié à l'opération de migration ?
- Quel point/composant du réseau a saturé ?

▶ Contexte organisationnel

- Est-ce que la charge du réseau est connue/surveillée ?

◆ **Gestion de projet : impossibilité de répondre à la contrainte**

▶ **Contexte métier :**

- La contrainte de disponibilité était-elle non révocable ?
- Le fournisseur avait-il connaissance de toutes les contraintes ?

▶ **Contexte applicatif :**

- L'ensemble des composants nécessaire à la construction du POC était-il maîtrisé du fournisseur ?
- Les interactions entre les différents composants étaient-elles maîtrisées ?

▶ **Contexte organisationnel :**

- La réponse à l'appel d'offre avait-elle été validée par les faiseurs ?
- Les compétences nécessaires à la réalisation était-elle présentes ?

Une infinité de cas...

◆ Une situation du SI en constante évolution :

- ▶ Tous les 5 ans, le coût du réseau baisse de 30 % pour un gain de performance de 30 %...
- ▶ La Loi de Moore est toujours valide...
- ▶ Le Cloud redéfinit le rôle des DSI au sein de l'entreprise...
- ▶ Tous les 5 ans, les IDE / méthodologie de programmation ont radicalement changées...
- ▶ ...

◆ Et étonnamment stable :

- ▶ La majorité (80%) des transferts de données sont réalisés en FTP...
- ▶ Les paradigmes de la programmation n'ont pas évolués depuis les années 70/80...
- ▶ ...

Quelques fondamentaux...

- ◆ Qui a une *solution* ?
- ◆ L'important est d'établir le lien entre les besoins (contraintes métiers) et les solutions (contraintes applicatives, technique et organisationnelles).
- ◆ C'est le travail de *l'architecte* qui a [Printz] :
 - ▶ *Une compréhension profonde des besoins présents et futurs [...] ;*
 - ▶ *Une connaissance approfondie des matériaux à assembler ainsi que de leur limites [...] ;*
 - ▶ *Une connaissance approfondie des procédés de construction (i.e. les méthodes du génie civil, la gestion de projet, l'acquisition)...*

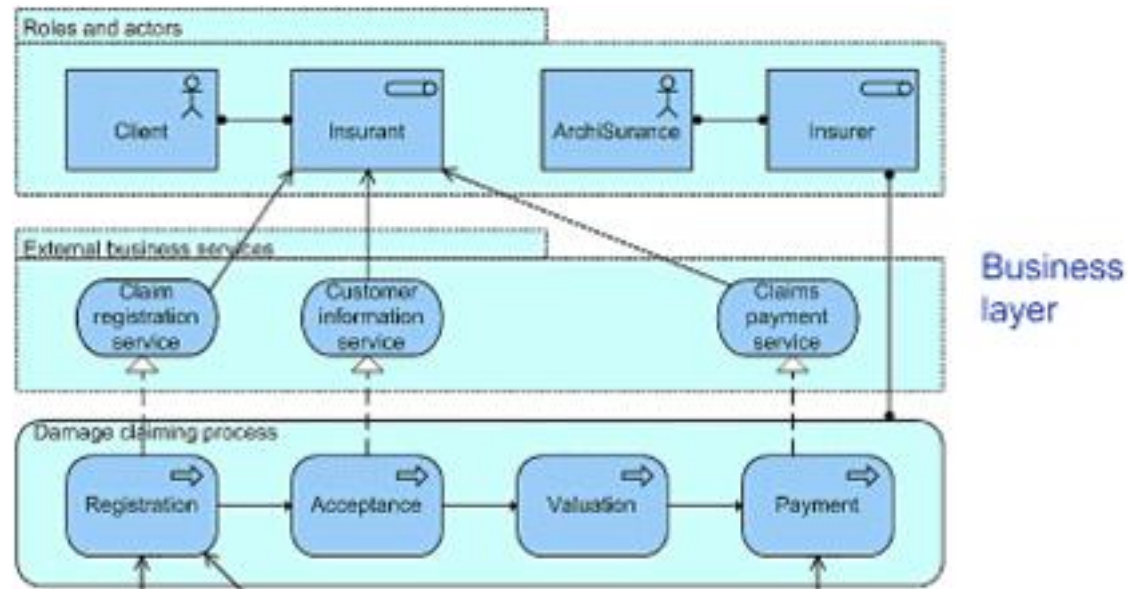
Plan du chapitre

1 Une infinité de cas...

2 Définitions / Les composants d'architecture SI

3 Démarche d'architecture SI

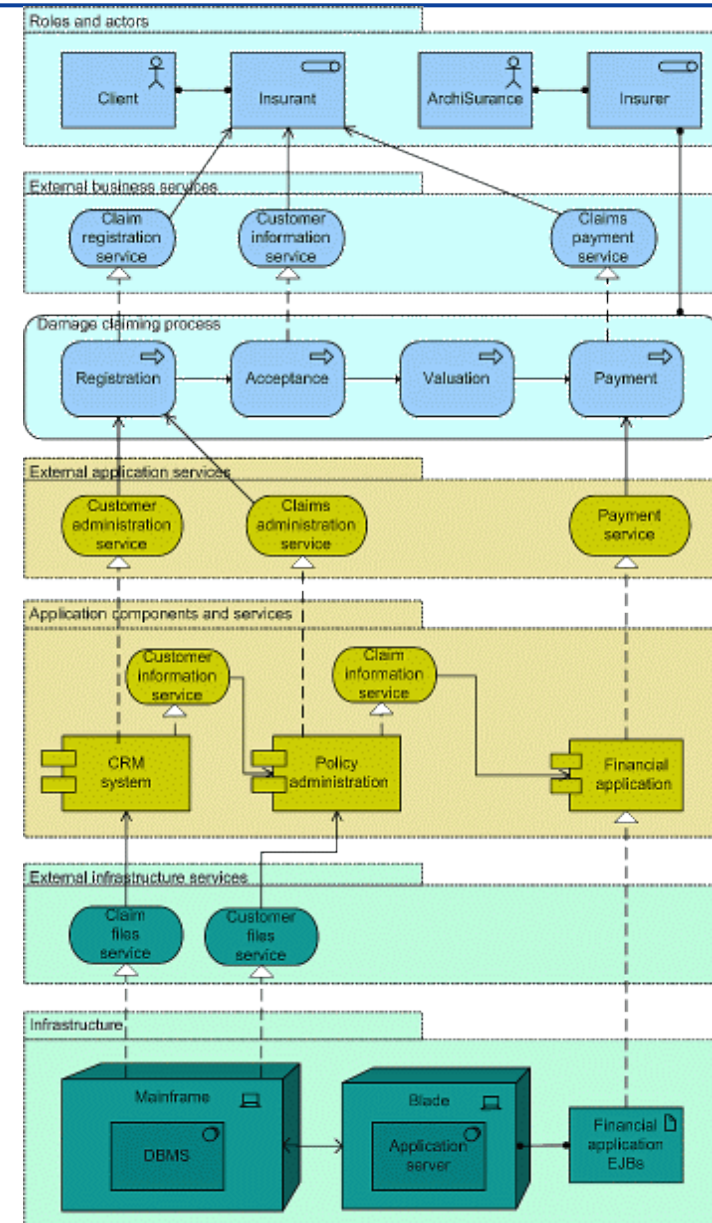
- ◆ Cas de l'assureur AlphaCentoris
- ◆ Il souhaite outiller l'étape de validation du processus de déclaration des accidents



Définition : composants de base du SI

◆ Les composants de base du SI sont :

- ▶ un ensemble *d'acteurs* qui réalisent des activités
- ▶ un ensemble *d'opérations* ou de *fonctions* applicables à des *données* réalisées par un ensemble *d'applications*
- ▶ un ensemble *d'échanges (de flux)*
- ▶ un ensemble de *données*
- ▶ un ensemble de *localisations*
- ▶ utilisant un ensemble de *briques d'infrastructure*



Définitions : architecture vs urbanisme

◆ Deux démarche s'opposent et se complètent :

▶ Bottom / up : *ce que le SI peut*

- **L'ensemble des contraintes techniques et opérationnelles** prescrivent et déterminent les solutions mises en œuvre
- C'est la démarche d'architecture du SI

▶ Top / down : *ce que le métier vœux*

- **Les nouveaux besoins métiers** prescrivent et déterminent les solutions mises en œuvre
- C'est la démarche d'urbanisation du SI

▶ **La bonne solution s'inscrit dans la double démarche d'urbanisation et d'architecture définissant « l'architecture d'entreprise »**

Définition : contraintes

- ◆ L'architecture va répondre de manière locale (à un besoin) et globale (SI)

- ▶ Aux contraintes (*exigences, besoins, ...*) « métiers »

	Fil conducteur
<ul style="list-style-type: none">• Fonctionnalités souhaitées• Performance• Niveau de service (SLO)• Sécurité présente• Coût permettant un ROI• Délais	<i>Outillage de la validation</i> <i>Les 11M d'assurés</i> <i>HO</i> <i>Présence d'ICS</i> <i>< 1M€</i> <i>Dès que possible (ASAP)</i>

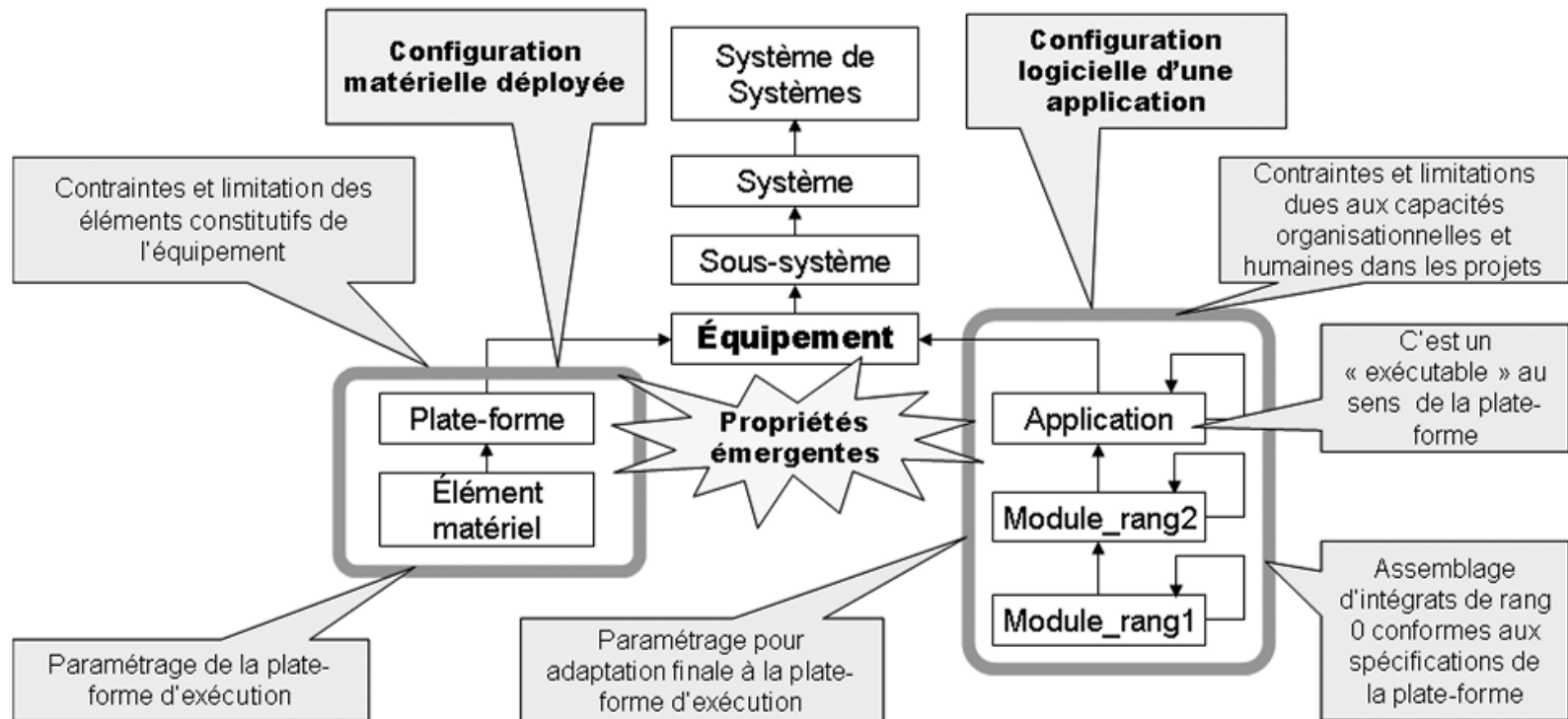
- ▶ Aux contraintes (*exigences, besoins, ...*) « opérationnelles et techniques »

<ul style="list-style-type: none">• Compétences disponibles et solutions maîtrisées• Performance• Niveau de service (SLA)• Sécurité respectée• Coût cohérent• Planning	<i>Évolution majeur</i> <i>Upgrade nécessaire</i> <i>À priori OK</i> <i>À renforcer</i> <i>2,5M€</i> <i>1 an</i>
---	---

◆ Quels sont les analyses de ces composants de base du système d'information ? en vision bottom/up on identifie...

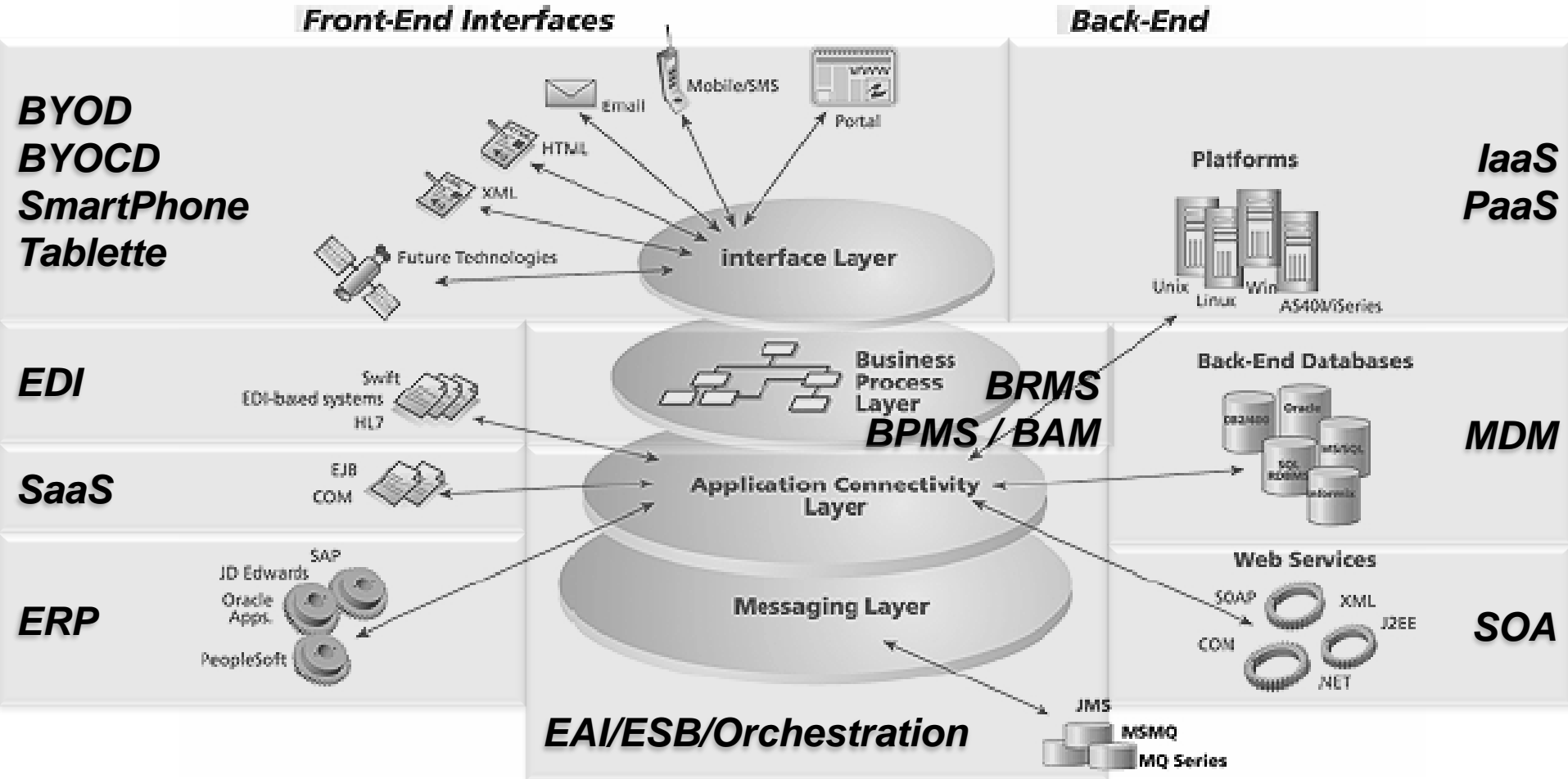
- ▶ **Briques d'infrastructure SI (réseau, virtualisation, IAM, ...)**
 - Plus ou moins partagés / externalisés
 - Constituent le « socle d'infrastructure SI » de l'entreprise
- ▶ **Application**
 - Qui s'intègre dans les briques d'infrastructures
 - Implémente une « logique métier »
- ▶ **Groupes d'applications (systèmes / chaîne applicative)**
 - Qui « collaborent » / « échangent » / ...
 - Outillent un processus métier

Définitions : de quoi est composé le SI ?



Définitions : oui... mais...

◆ Le SI n'est plus aussi simple :



◆ Une architecture opérante résulte :

▶ D'une démarche d'architecture

- Pendant la construction des systèmes/applications
- Avec les bons outils (de modélisation) d'architecture

▶ D'une utilisation *à bon escient* des briques qui composent le SI

- Comprendre leurs usages
- Et leur niveau de maturité

▶ Dans le respect de l'ensemble des contraintes

Plan du chapitre

1 Une infinité de cas...

2 Définitions / Les composants d'architecture SI

3 Démarche d'architecture SI

3.1 Structuration en vue

3.2 Design par contraintes

3.3 Organisation de l'architecture

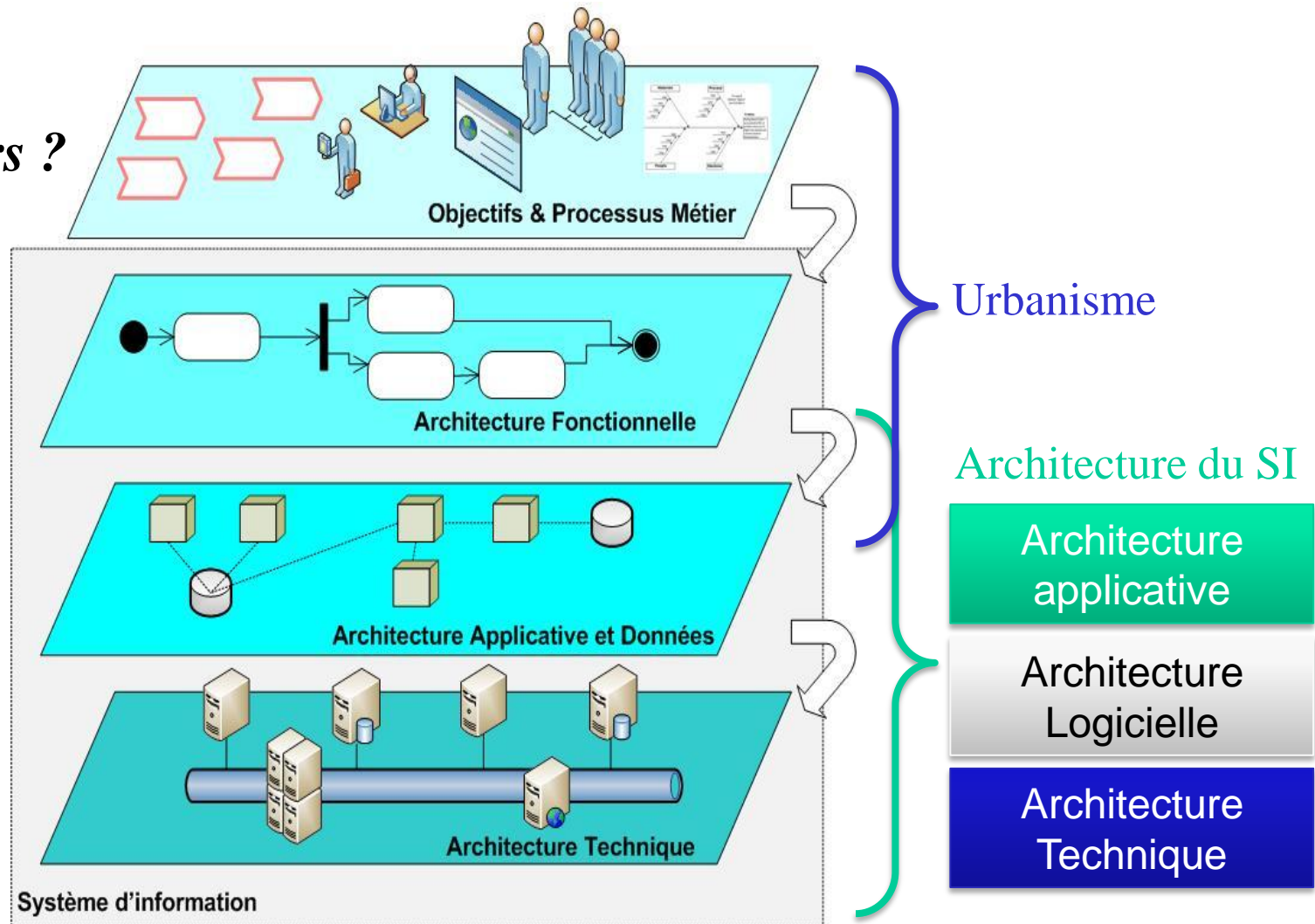
Démarche d'architecture SI : structuration en vues

Quels métiers ?

Quoi?

Comment?

Avec quoi?



◆ Architecture Applicative

- ▶ Elle structure le SI en blocs applicatifs communicants
- ▶ Elle décrit sous l'angle technique les applications, les flux et les messages échangés entre applications

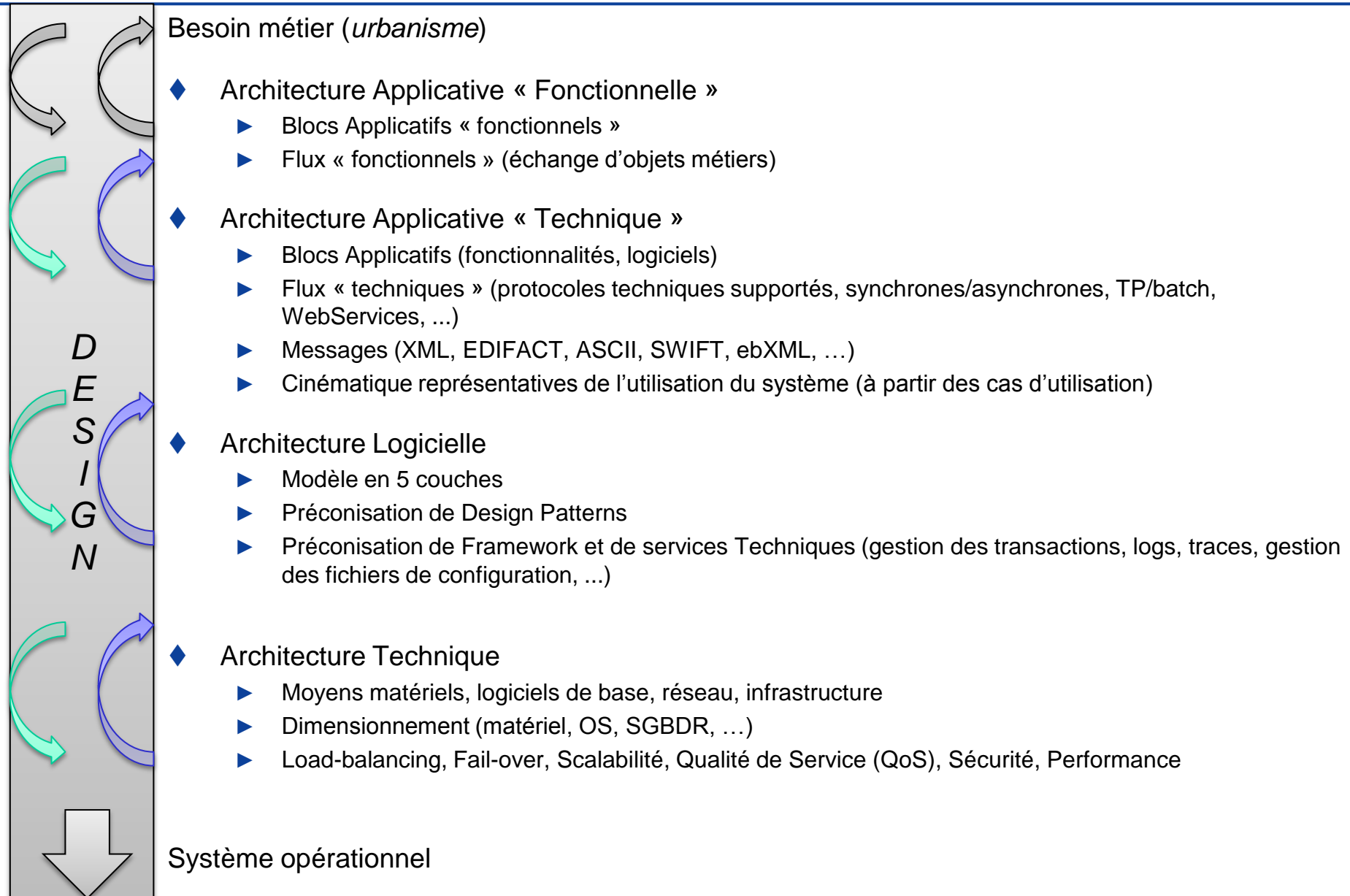
◆ Architecture Logicielle

- ▶ Elle structure les solutions en œuvre dans une application les mieux à même de répondre aux spécifications fonctionnelles
- ▶ Elle structure et décompose de façon logique chaque partie de l'application via les notions et concepts de découpage en couches, composants, framework et design patterns

◆ Architecture Technique

- ▶ Elle structure les solutions d'infrastructure technique du SI
- ▶ Elle décrit et organise
 - les différents moyens matériels (serveur, poste client ...)
 - les logiciels de base (systèmes d'exploitation, SGBD, AGL ...)
 - les moyens de communication (réseaux, middleware...)

Démarche d'architecture SI : méthodologie globale



Démarche d'architecture SI : livrables

- ◆ **Les livrables des phases d'architecture (logique et technique) sont constitués d'une partie modélisation, une partie documentaire et d'une partie opérationnelle :**
 - ▶ **La modélisation dans le référentiel d'architecture d'entreprise suivant les démarches (TOGAF, Archimate, ...) en place dans l'entreprise**
 - ▶ **Le dossier/les vues d'Architecture de haut niveau (HLD : High Level Design) :**
 - Architecture applicative,
 - Architecture logicielle,
 - Architecture physique
 - ▶ **Les directives opérationnelle, technique et méthodologique précises (LLD : Low Level Design) :**
 - Description précise des paramétrages des environnements de développement, de tests, d'intégration, de production,
 - Règles de nommage,
 - Frameworks et modes d'implémentation,
 - Services « métier » communs (parseur de messages, machine à états, ...),
 - Services techniques communs (IHM, logs...),
 - Guides méthodologiques

Plan du chapitre

1 Une infinité de cas...

2 Définitions / Les composants d'architecture SI

3 Démarche d'architecture SI

3.1 Structuration en vue

3.2 Design par contraintes

3.3 Organisation de l'architecture

◆ Contraintes métier

- ▶ Répondre à un besoin ponctuel
- ▶ Être en adéquation avec la stratégie d'entreprise

◆ Contraintes SI

- ▶ Agilité
- ▶ Compétences

◆ Et en bonus :

- ▶ Sécurité (SI et Métier)
- ▶ Budgétaire

La responsabilité de l'adéquation entre les besoins ponctuels des métiers et la stratégie globale de l'entreprise est un des rôles de l'urbaniste. L'architecte travail à questionner les contraintes métiers lorsque les solutions SI ne peuvent pas les respecter.

Design par contraintes : les principales contraintes SI

- ◆ **L'architecture doit s'adapter au contexte de l'entreprise :**
 - ▶ Les politiques (en particulier sécurité)
 - ▶ Le référentiel achat
 - ▶ Le référentiel des solutions techniques maîtrisées (OS, ...)
 - ▶ Le référentiel d'architecture,
 - ▶ Le socle d'infrastructure (réseau, sécurité, IAM, fondations, ...)
 - ▶ Les contraintes d'ergonomie (charte graphique, ...)
 - ▶ Le référentiel des méthodes d'implémentation (frameworks, patterns, ...)

- ◆ **Et répondre aux exigences de l'application cible :**
 - ▶ Fonctionnelle,
 - ▶ De performances : temps de réponse moyens, volumétrie / fréquence, évolution dans le temps, ...
 - ▶ De disponibilité de l'application (SLA, SLO, DIMA, PDMA, ...),
 - ▶ D'évolutivité (lotissement, ...)

◆ Les contraintes organisationnelles (compétences) :

- ▶ **Adéquation entre la cible et la culture de développement des équipes du projet : selon qu'elles sont familières des notions de conception et de développement orientées objet ou bien teintées développement procédural**
- ▶ **Adéquation entre le fonctionnement actuel de la production et la cible : fil de l'eau versus batchs, spécifique vs sur étagère, ...**
- ▶ **Adéquation entre les contraintes posées par l'architecture sur les attendus et les compétences des Maîtrises d'Ouvrage et des métiers (production de spécification d'échanges, prise en compte des contraintes d'opérabilité, ...)**

Design par contraintes : c'est quoi un bon design ?

◆ Il existe de nombreuses façons de valider / qualifier une architecture :

- ▶ **Adéquation avec le besoin**
 - Couverture fonctionnelle de la solution
- ▶ **Capacité d'Agilité / Extensibilité :**
 - Capacité à intégrer rapidement de nouveaux flux et/ou de nouveaux applicatifs : intégration d'un nouveau canal de distribution, sous-traitance d'une activité, etc.
 - Capacité à supporter plusieurs modes d'organisations
- ▶ **Respect des standards applicatifs et techniques de l'entreprise**
 - L'architecture utilise-t-elle les fonctions transverses mises en place au niveau SI, comme les référentiels, la sécurité, les plates-formes d'échanges, les services communs ?
 - L'architecture technique s'insère-t-elle dans l'univers d'exploitation connu ou nécessite-t-elle de nouvelles technologies ?
- ▶ **Evolutivité :**
 - Capacité à intégrer de nouveaux utilisateurs, des référentiels plus larges, montée en charge, etc.
- ▶ **Sécurité :**
 - Mesure de la sécurité globale du système : sécurisation des échanges, difficulté d'intrusion, authentification des acteurs, etc.
- ▶ **Coûts :**
 - Mesure du rapport qualité/prix (benchmark par rapport à d'autres constructions)
 - Mesure de la rentabilité (ROI)
- ▶ **Exploitabilité :**
 - Réponse effective aux contraintes de disponibilité

Plan du chapitre

1 Une infinité de cas...

2 Définitions / Les composants d'architecture SI

3 Démarche d'architecture SI

3.1 Structuration en vue

3.2 Design par contraintes

3.3 Gouvernance de l'architecture

Gouvernance de l'architecture

Initiate
émergence

Define
définition

Design
conception

Build
codage

Test

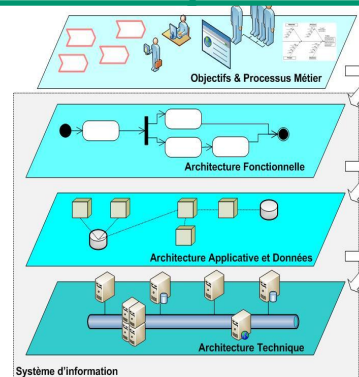
Deploy
intégration

Run

Phase de définition de la solution, des méthodes d'implémentation et d'intégration.
Gouvernance des choix en fonction des contraintes stratégiques.

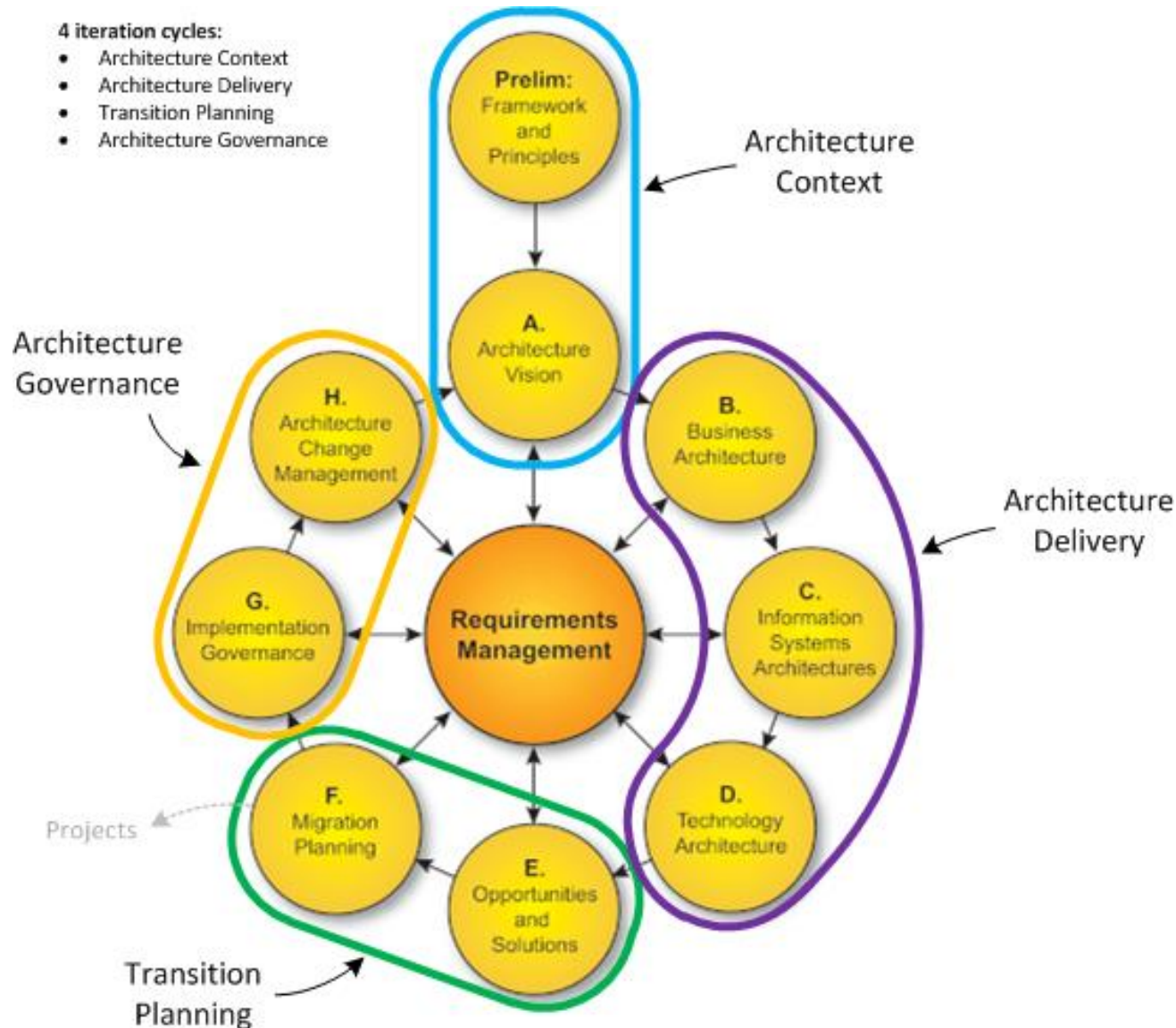
Phase de construction (implémentation).
Gouvernance des adaptations en confrontation au réel (gestion des problèmes et des incidents).

Architecte concepteur



Architecte intégrateur

Gouvernance de l'architecture : cycle TOGAF



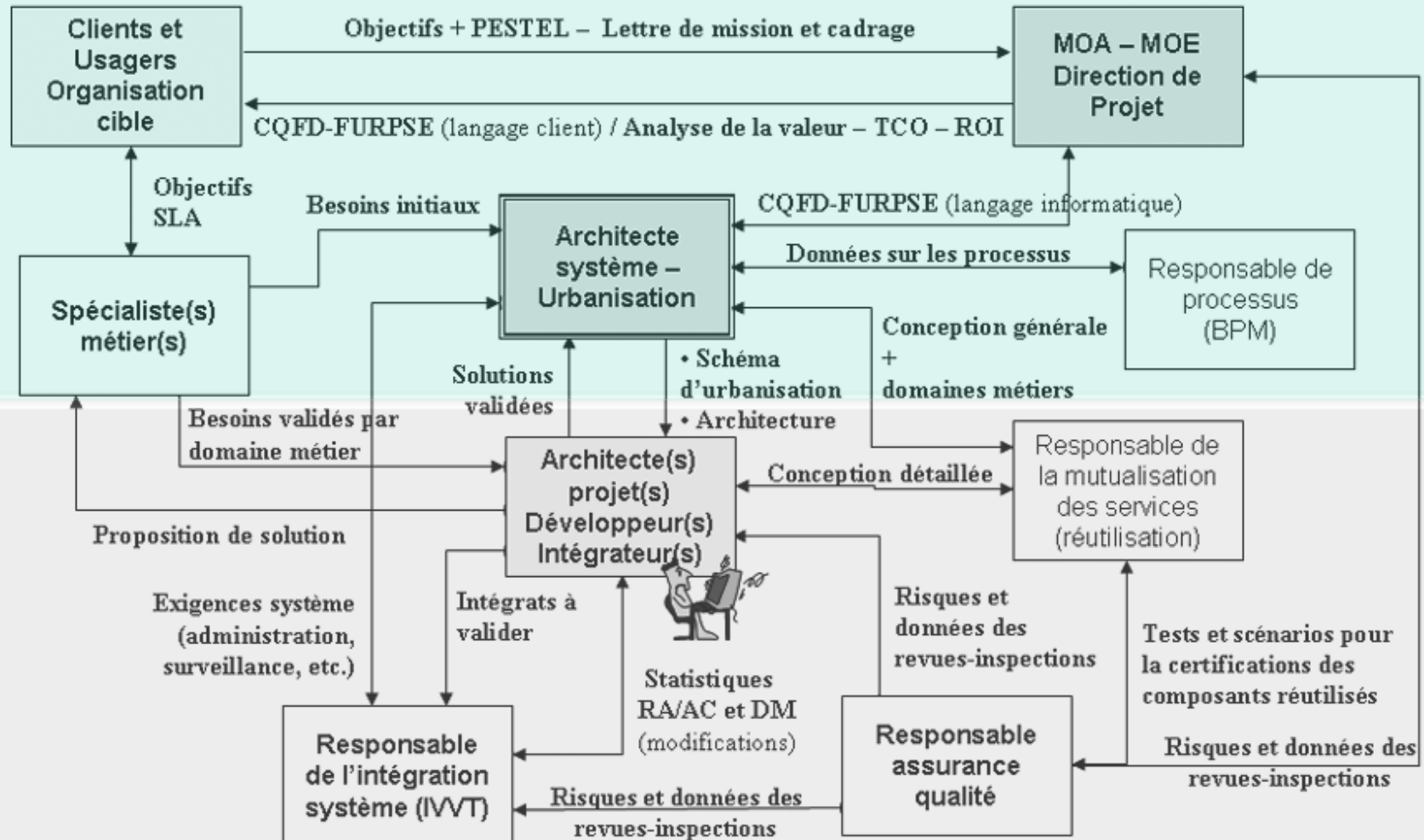
- ◆ De par sa position centrale sur l'ensemble des phases de conception et de réalisation, ayant une vision globale contraintes, l'architecte définit la composition de la solution, qui la modifie et l'affine par itérations successives : re-découpage / fusion de briques applicatives, application d'un pattern, mutualisation de flux...
- ◆ Il a pour rôle de:
 - ▶ Recenser les besoins techniques (analyse de l'existant, contraintes, besoins exprimés)
 - ▶ Définir les principes directeurs de l'architecture
 - ▶ Elaborer l'architecture applicative, logicielle et physique
 - ▶ Argumenter ses choix technologiques
 - ▶ Identifier les besoins en produits tiers et frameworks techniques
- ◆ En respect des phases du projet.

◆ Best Practices

- ▶ **Prendre connaissance du fonctionnel en récupérant les informations des analystes et avec les urbanistes**
- ▶ **Assister le concepteur sur les premières modélisations**
 - L'architecte définit avec les équipes de conception :
 - La modélisation du découpage du modèle en couches
 - La définition des stéréotypes et des contrats entre les couches...
- ▶ **Assister le développeur sur le codage des premiers modules applicatifs**
 - L'architecte définit avec les équipes de développement :
 - Le découpage du modèle en couches
 - Les framework à mettre en œuvre
 - Les design patterns à utiliser
 - Les règles de nommages des packages...
- ▶ **Participer / piloter les phases d'intégration**
- ▶ **Participer / piloter les benchmarks**

Gouvernance de l'architecture : positionnement de l'architecte [Prinz]

Quoi



Comment

Gouvernance de l'architecture : positionnement de l'architecte

exemple d'organisation d'un grand groupe : analyse critique ?

