



ARCHITECTURE LOGICIELLE

1

PLAN DU COURS

- INTRODUCTION AUX ARCHITECTURES LOGICIELLE
 - PROBLÉMATIQUE
 - DÉFINITION
 - ARCHITECTURE VS CONCEPTION
 - QUI S'EN CHARGE?
 - CRITÈRES D'ÉVALUATION
 - QUELQUES PATTERNS ARCHITECTURAUX



Problématique

- Complexité des applications:
 - Distribution des applications en plusieurs machines,
 - Multitude des langages et des plateformes
- Nécessité de la réutilisation des applications existantes
- Evolution des systèmes



Solution

Définir une architecture logicielle pour:

- Améliorer la qualité et la production du code
- Réduire les coûts
- Simplifier le travail du programmeur
- Réutiliser au mieux le logiciel

Qu'est ce qu'une architecture logicielle?

L'architecture d'un système logiciel se préoccupe fondamentalement de **l'organisation d'un système** en ses **constituants** et leurs **interrelations** afin de réaliser un objectif donné.

Architecture  Conception

ARCHITECTURE

- Répond à **Quoi?**
- Packages, composants et relations
- Pertinente pour les objectifs opérationnels et exigences non fonctionnelles

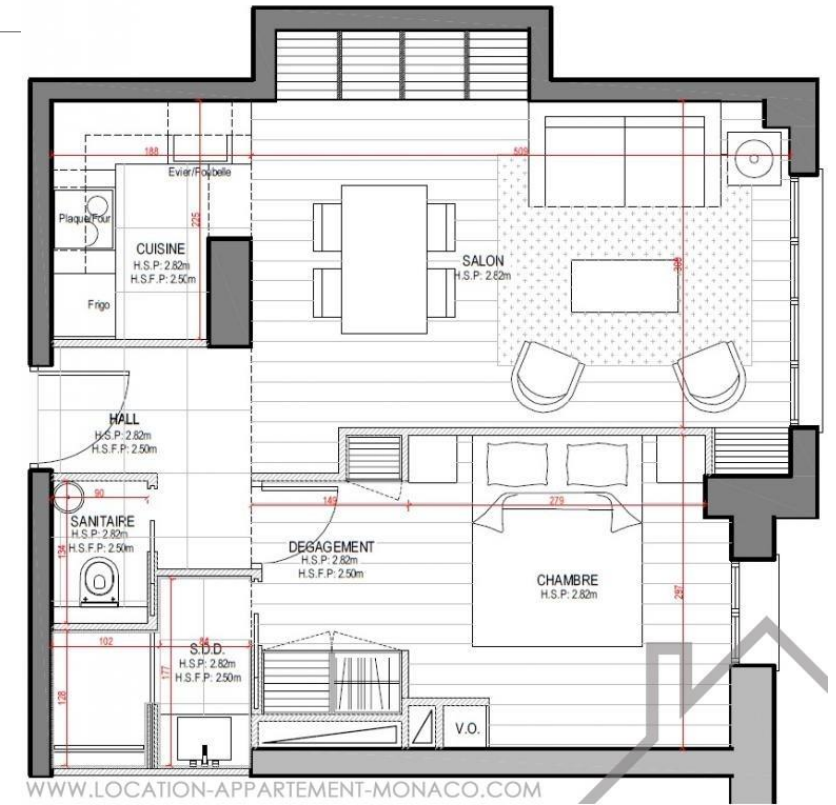
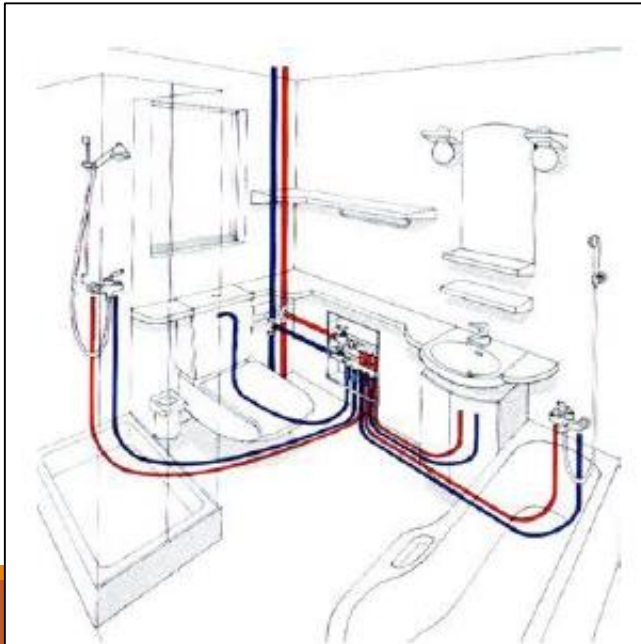
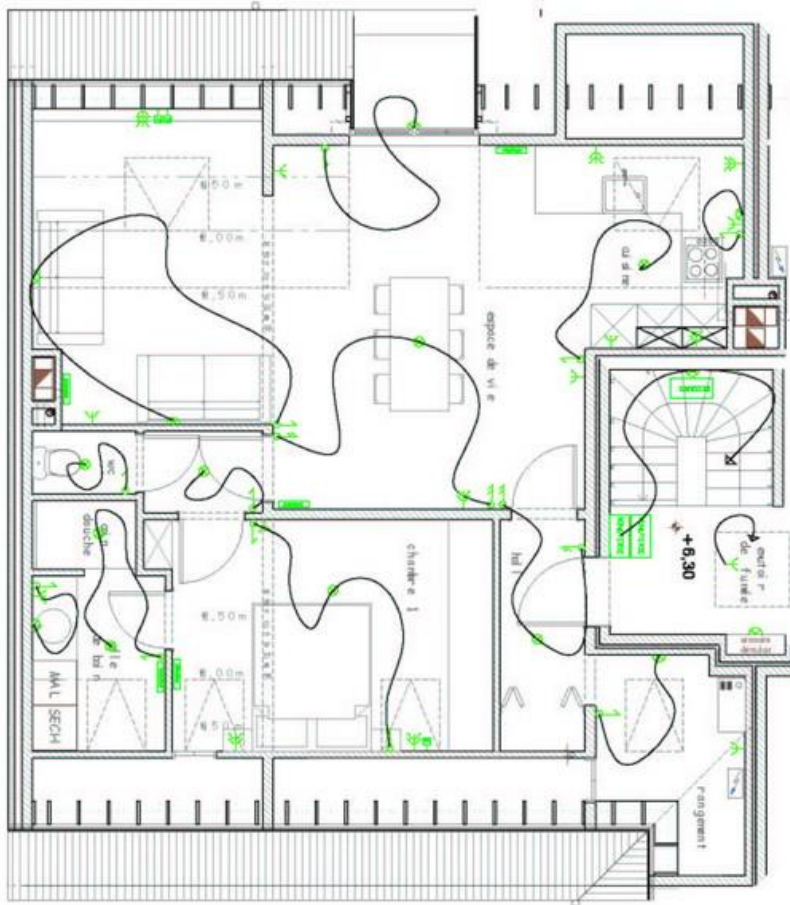
CONCEPTION

- Répond à **Comment?**
- Classes, méthodes, Fonctions, ...
- Pertinente pour les exigences fonctionnelles

L'architecture dans l'immobilier



La conception dans l'immobilier



Architecture



Conception

«On veut une couche de GUI, une couche de calcul et une couche de stockage des données.»

Architecture



Conception

«Toutes les nouvelles classes doivent étendre
l'interface *Application* »

Architecture



Conception

«L'application sera disponible comme un service
déployé en 'Cloud'»

Architecture



Conception

«On a besoin d'une base de données NoSQL
avec une disponibilité élevée »

Architecture



Conception

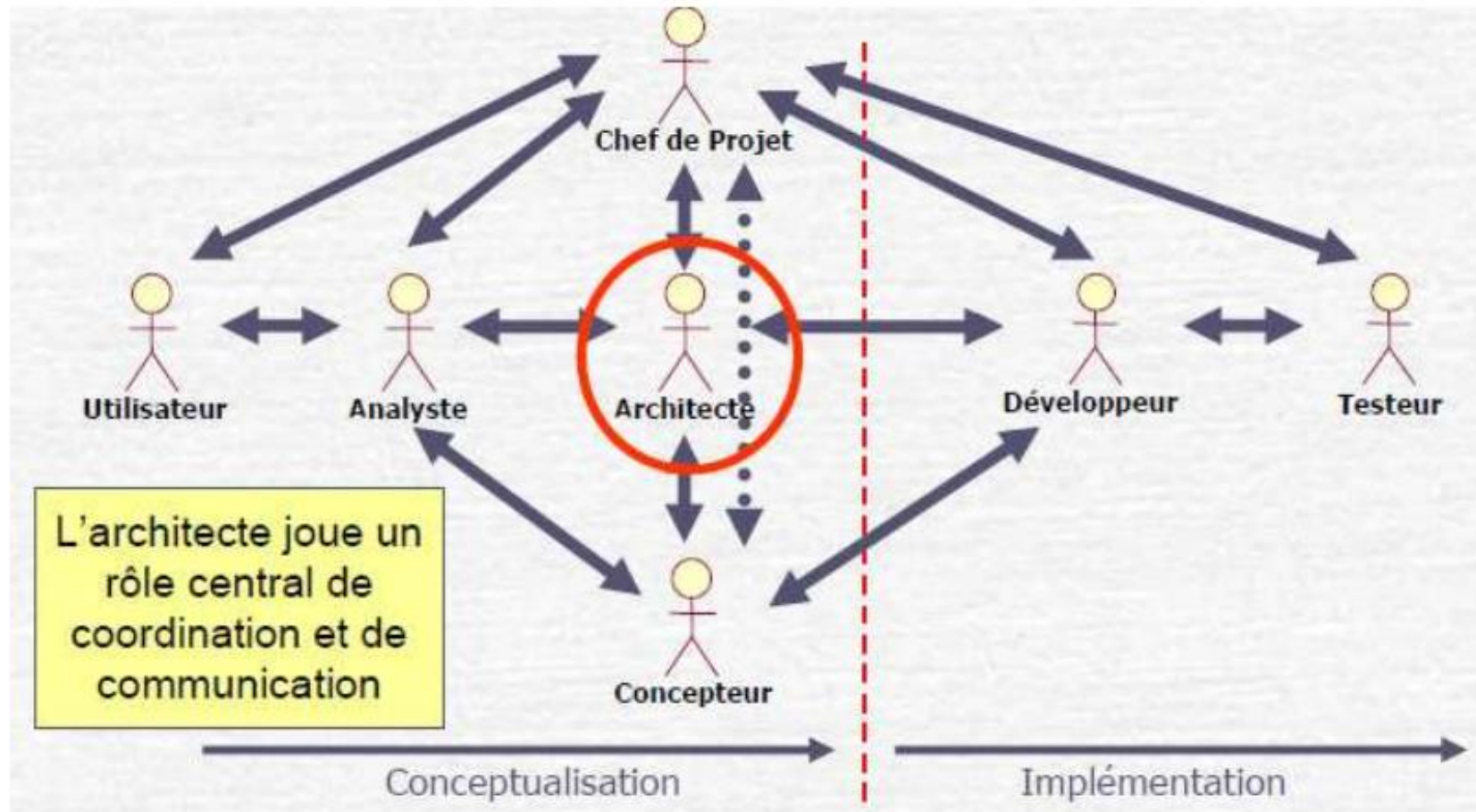
«Une méthode prend le type d'un objet comme paramètre et retourne une instance de ce type en appelant le constructeur privé de la classe correspondante»

Architecture



Conception

Qui s'en charge?



Critères d'évaluation

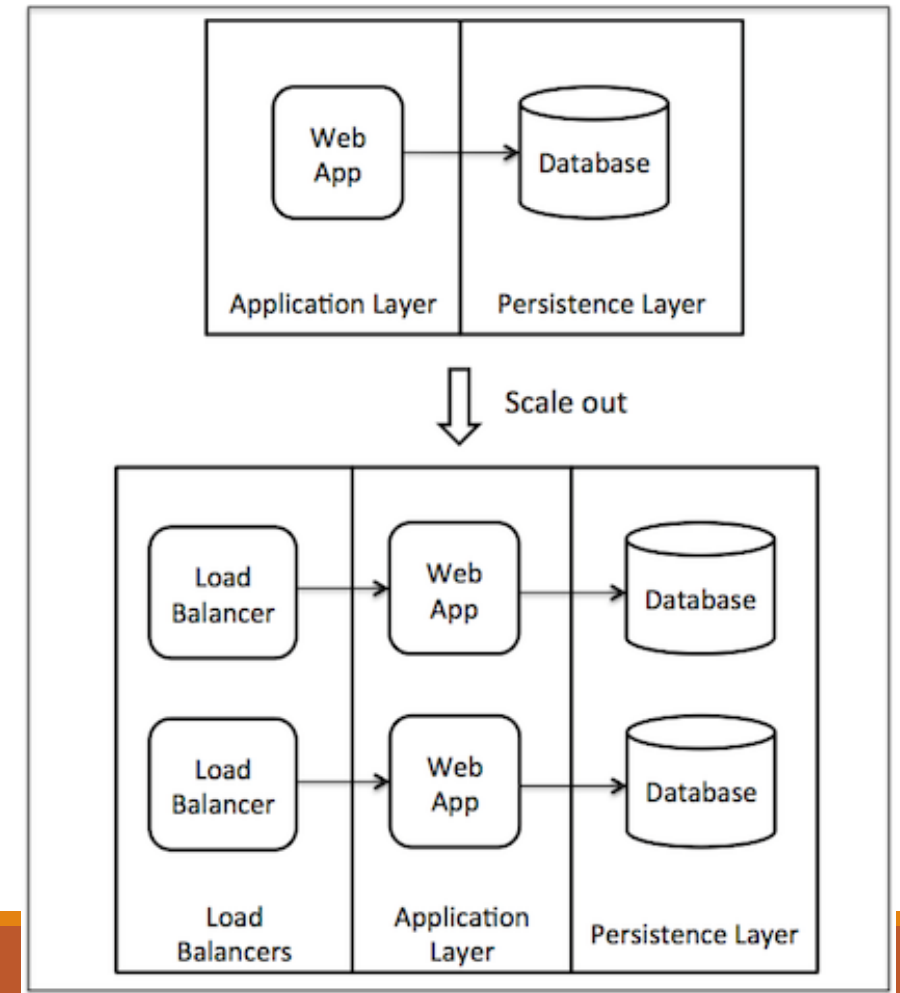
- Plusieurs critères peuvent guider le choix de l'architecture logicielle:
 - Déploiement
 - Testabilité
 - Performance
 - Evolutivité
 - Développement

Critères d'évaluation: Performance

- Le temps de réponse (utilisateur, réseau, requête),
- La capacité à supporter n utilisateurs simultanément,
- La consommation de ressources (mémoire, processeur, disque et réseau),
- La stabilité (fonctionnement nominal sans erreur ni compromission des données),

Critères d'évaluation: Evolutivité

- Gérer une charge croissante de travail
- Gérer un nombre d'utilisateurs croissant
- Gérer un volume de données croissant



Quelques exemples d'architectures

