

Systems Architecting

Abdellatif MEGNOUNIF

Partie 3: Modèles et Modélisation

Chap. 9

Modèles de Représentation et architecting des systèmes



1. Introduction: Rôles, Vues et Modèles

- ❑ Les modèles sont les moyens de communication entre clients, constructeurs et utilisateurs.
- ❑ Les modèles sont les langages de l'architecte.
- ❑ Les modèles permettent, guident et aident l'évaluation de la construction des systèmes au fur et à mesure de leur développement progressif.
- ❑ Après la construction du système, les modèles, des simulateurs aux manuels de fonctionnement aident à décrire et diagnostiquer son fonctionnement.
- ❑ L'architecte doit maîtriser tous les langages le long du processus du développement du système.
- ❑ Les langages des concepteurs du système, des constructeurs, des certificateurs, des distributeurs et des utilisateurs...
- ❑ Les modèles les plus importants sont ceux qui définissent les exigences d'acceptation critiques du client et l'ensemble de la structure du système.

1. Introduction: Rôles, Vues et Modèles (suite)

- ☐ Parce que l'architecte est responsable de la faisabilité du système totale, les portions critiques peuvent inclure les modèles hautement détaillés des composants dont le succès dépendra
- ☐ Les modèles sont classés par leur rôle ou bien par leur contenu.
- ☐ Le rôle est important pour relier les modèles aux tâches et responsabilités non seulement des architectes mais d'autres dans le processus de développement.
- ☐ une importance particulière pour les architectes sont des méthodes de modélisation qui lient des modèles distincts en un tout cohérent.

2. Rôles des modèles

- ❑ En architecting des systèmes, les modèles ont plusieurs rôles:
 1. Communication avec le client, les utilisateurs et les constructeurs.
 2. Maintenance de l'intégrité du système à travers une coordination des activités du design.
 3. Assister le design en fournissant des modèles, organiser et enregistrer les décisions.
 4. Exploration et manipulation des paramètres et caractéristiques de la solution; guider et enregistrer l'agrégation et la décomposition des fonctions, composants et objets du système.
 5. Prédiction de la performance et identification des éléments du système critiques.
 6. Fournir les critères d'acceptation pour certification pour utilisation.

2. Rôles des modèles (suite)

- ❑ Ces rôles dépendent l'un de l'autre. Mais le plus important est de communiquer.
- ❑ Les modèles du système sont le moyen de communication de l'architecte avec le client, les utilisateurs, les constructeurs et d'autres groupes qui sont intéressés.
- ❑ Les modèles utilisés pour communication deviennent une documentation de décisions et designs et donc un véhicule pour maintenir l'intégrité du système.
- ❑ La communication avec **le client** a 02 buts: (1), l'architecte doit déterminer les objectifs et contraintes du client. (2) l'architecte doit assurer que le système à construire reflète les jugements de valeur du client où l'accomplissement parfait de tous les objectifs est impossible.

2. Rôles des modèles (suite)

- ❑ La communication **avec l'utilisateur** est similaire. Ça nécessite une explication des besoins et la comparaison de systèmes possibles pour rencontrer ces besoins.
- ❑ Si le client est lui-même utilisateur, le processus est simplifié.
- ❑ Si le client est différent de l'utilisateur, il peut y avoir un conflit dans les besoins et les contraintes. L'architecte est dans une position de conciliateur entre les 02.
- ❑ Dans une communication à 02 sens avec **le constructeur**, l'architecte recherche à assurer que le système sera construit comme il est conçu et que l'intégrité du système est maintenue.
- ❑ En plus, l'architecte doit apprendre du constructeur les contraintes techniques et les opportunités qui sont cruciales pour assurer un design satisfait et faisable.
- ❑ Modèles liant le client au constructeur peuvent aider à fermer les itérations de la capabilité technique du constructeur aux objectifs du client.

3. Modèles, points de vue et vues.

- ❑ IEEE définit ces termes comme suit:
- ❑ **Un modèle** est une approximation, une représentation et une idéalisation d'aspects sélectionnés de la structure, du comportement, du fonctionnement ou bien d'autres caractéristiques d'un processus du monde réel, d'un concept ou bien d'un système.
- ❑ **Une vue** est une représentation du système d'une perspective des préoccupations ou des pbs liés.
- ❑ **Un point de vue** est un modèle, un style ou bien une spécification pour la construction d'une opinion.

3. Modèles, points de vue et vues. (suite)

- ❑ **Un modèle** est juste une représentation de certains aspects de l'architecture du système.
- ❑ Les langages de modélisation ont un vocabulaire et une grammaire. Les mots font partie du modèle et la grammaire définit comment les mots peuvent être liés.
- ❑ En plus, le langage de modélisation doit avoir une méthode d'interprétation et les modèles produits doivent avoir un sens à l'intérieur d'un domaine spécifique.
- ❑ Ex: la méthode de diagramme en blocs, les mots sont les types de blocs, de lignes et les connections aux choses réelles.
- ❑ Une méthode physique doit avoir une correspondance aux choses physiques identifiables.
- ❑ Une technique de diagramme fonctionnel doit avoir une correspondance aux fonctions que le système doit prendre en charge.

3. Modèles, points de vue et vues. (suite)

- ❑ **Une vue** est juste une collection de modèles qui partage la propriété qu'ils sont pertinents pour les mêmes préoccupations des intervenants du système.
- ❑ Ex: une vue fonctionnelle collecte les modèles qui représentent une fonction du système.
- ❑ Une vue d'objectifs collecte les modèles qui définissent les objectifs à atteindre en construisant le système.
- ❑ L'idée de la vue est nécessaire parce que les systèmes complexes tendent à avoir des modèles complexes et exigent un niveau élevé d'élément d'organisation.
- ❑ La vue c'est comme chez l'architecte classique. Il produit les élévations, les plans de planchers et autres représentations qui montrent le système d'une certaine perspective.
- ❑ **Un point de vue** est une abstraction de vues à travers plusieurs systèmes. C'est important uniquement dans la définition de normes pour une description architecturale.

4. Classification des modèles par vue

- ❑ Une vue décrit un système par rapport à un groupe d'attributs ou de préoccupations.
- ❑ Le groupe de vues choisi pour décrire un système est variable.
- ❑ Un bon groupe de vues doit être complet (couvrir toutes les préoccupations des intervenants de l'architecte) et généralement indépendant (capture différentes pièces d'information).

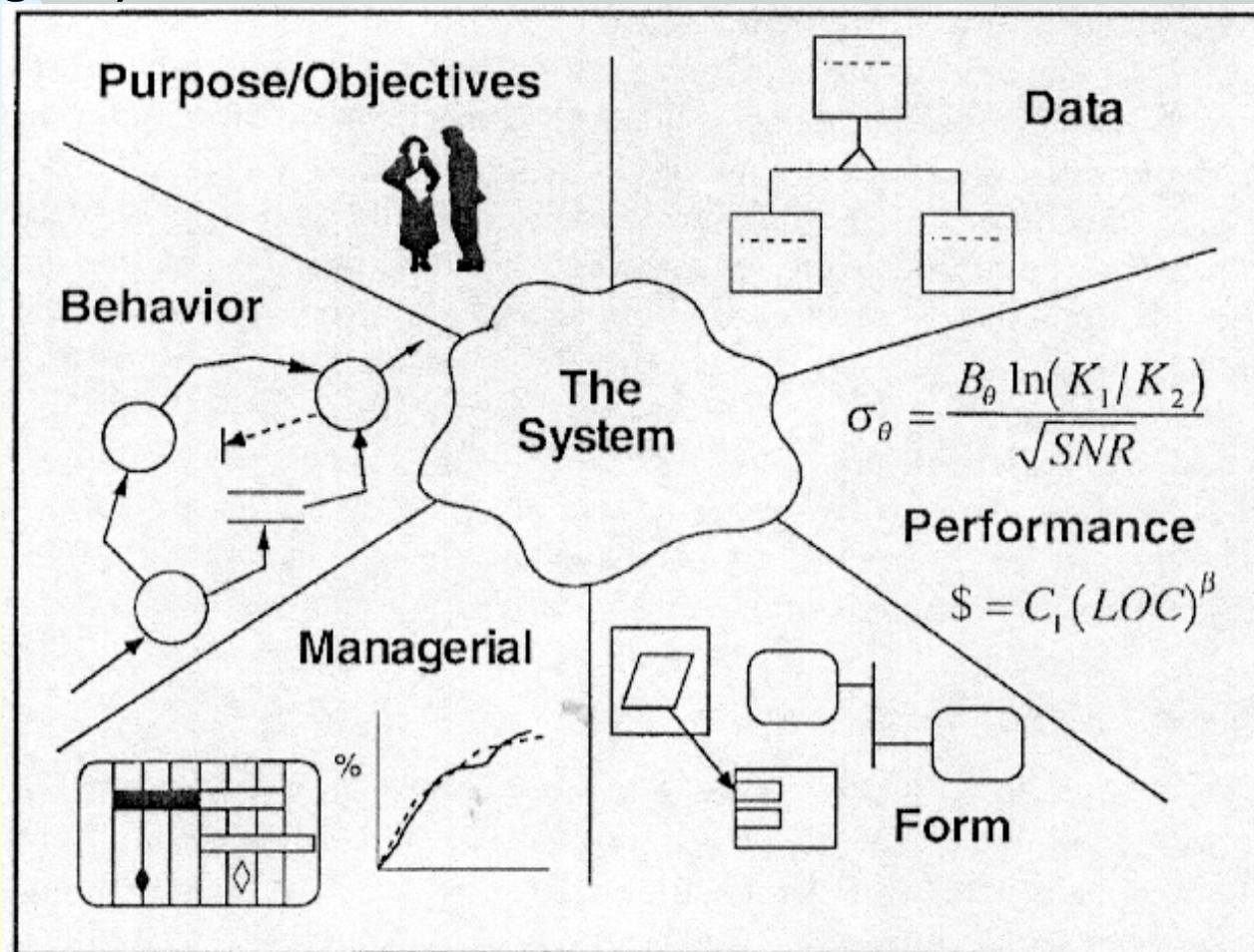
Vues architecturales principales

Perspective ou vue	Description
But / Objectif	Qu'est ce que veut le client
Forme	Que sera le système
Comportemental ou fonctionnel	Que fait le système
Exigences ou objectifs de la performance	Comment effectivement le système le fait
Données	L'information retenue dans le système et ses inter-relations
Managérial	Le processus par lequel le système est construit et managé.



4. Classification des modèles par vue (suite)

- ❑ Un système peut être projeté en n'importe quelle vue, de plusieurs manières.
- ❑ Une exemple de projection en vues et la collection de modèles par vues (figure)



4. Classification des modèles par vue (suite)

- ☐ Chaque système a un certain comportement, a une forme physique, retient des données, etc...
- ☐ Les vues sont composées de modèles.
- ☐ Pas toutes les vues sont importantes de façon égale pour le succès du développement du système.
- ☐ Le groupe de vues aussi n'est pas constant dans le temps.
- ☐ Bien que n'importe quel système peut être décrit dans chaque vue, la complexité et la valeur de chaque description de vue peut différer considérablement.
- ☐ Chaque classe de systèmes met l'accent sur des vues particulières et a favorisé des méthodes de modélisation ou bien de représentation au sein de chaque vue.
- ☐ L'architecte doit déterminer quelles vues sont importantes pour le système et son environnement et qu'il soit expert dans les modèles.

4. Classification des modèles par vue (suite)

- ☐ Bien que les vues sont choisies indépendantes de façon raisonnable, il existe un lien vaste entre les vues. Ex: les aspects de comportement du système ne sont pas indépendantes de la forme du système (il ne peut produire un comportement désiré que si la forme du système peut le supporter).
- ☐ Ce lien est similaire, de façon conceptuelle, à la vue de face et de coté qui sont liées (les 02 montrent des hauteurs verticales), bien qu'elles sont des observations obtenues de directions orthogonales).
- ☐ On va essayer de décrire les modèles utilisés pour représenter un système dans chacune des vues du tableau.

4. Classification des modèles par vue (suite)

4.1 Modèles des objectifs et but

- ❑ Les systèmes sont construits pour des fins utiles, **que veut le client.**
- ❑ Le premier rôle (et le plus important) de l'architecte est de faire correspondre la désirabilité des buts avec la faisabilité pratique du système pour répondre à ces fins.
- ❑ Avec le client, l'architecte essaye d'abord d'identifier les objectifs et priorités du système (certaines sont mesurables d'autres non) (une belle maison??).
- ❑ Il faut donc des modèles pour expliquer les objectifs non mesurables.
- ❑ La modélisation commence par retraiter et itérer les objectifs sans contraintes initiaux du langage du client jusqu'à avoir une méthodologie et un langage de modélisation, le 1^{er} pas important proche du développement ingénierie.

4. Classification des modèles par vue (suite)

4.1 Modèles des objectifs et but (suite)

- ❑ Les objectifs de comportement sont redéfinis dans un langage de modélisation de comportement.
- ❑ Les exigences de performance sont formulées comme modèles de satisfaction mesurables.
- ❑ Certains objectifs peuvent être traduits directement en formes physiques, d'autres en modèles de forme qui doivent être exposés par le système.
- ❑ Des objectifs complexes exigent plusieurs étapes de raffinement et peuvent aboutir à des caractéristiques ou comportements différents de leur état original.
- ❑ Pour savoir la puissance du design et de la méthode d'implémentation c'est par la possibilité de retenir le langage original. Mais ce n'est pas facile, il faut avoir un grand bagage de connaissances.

4. Classification des modèles par vue (suite)

4.2 Modèles de la forme

- ❑ Les modèles de la forme représentent des éléments identifiables physiquement de, et les interfaces à, ce que va être construit et intégré pour atteindre les objectifs du client.
- ❑ Les modèles de forme varient largement dans leur degré d'abstraction et de rôle.
- ❑ Les 02 catégories de modèles de forme les plus utilisés en architecting sont les **modèles d'échelles** (scale models) et les **diagrammes en block**.

Modèles d'échelles

- ❑ Les modèles d'échelles sont largement utilisés pour la communication du constructeur et du client et peut fonctionner comme partie de la modélisation de la performance et du comportement.
- ❑ Ex: modèles en construction, en automobiles (formule 1), bateaux (courses de voiles...), prototypes pour les logiciels (log de démonstration...).



4. Classification des modèles par vue (suite)

4.2 Modèles de la forme (suite)

Diagramme en blocks

- ❑ Les diagrammes en block sont omniprésents dans l'industrie électronique.
- ❑ Pour être un modèle de forme, les éléments du diagramme en block doivent correspondre à des éléments du système identifiables physiquement.
- ❑ Attention, souvent dans ce type de diagramme on oublie la logique de transmission des données. Le diagramme peut montrer le sens de transmission d'un module A vers un module B mais ne précise pas qui contrôle la transmission.

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels)

- ☐ Définissent des modèles spécifiques de comportement par le système.
- ☐ Ce sont des modèles de ce que va faire le système (comment il se comporte) à l'opposé de ce qu'est le système (modèles de forme).
- ☐ La nécessité de modèles de comportement augmente au fur et à mesure que le système devient intelligent.
- ☐ La détermination du niveau de détails ou de rigueur dans la spécification de comportement exigée pendant l'architecing est un choix important.
- ☐ Moins de détails peut conduire à la non compréhension du comportement par le client.
- ☐ Plus de détails peut rendre les spécifications incompréhensibles (peut conduire à des pbs similaires).

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

- ❑ Quel sera le niveau de raffinement du modèle de comportement ?
- ❑ Le meilleur est de se concentrer sur les exigences d'acceptation du système.
- ❑ Demander quels sont les attributs de comportement du système, le client va demander d'être certifiés avant acceptation et déterminer à travers quels tests ils doivent être certifiés.
- ❑ Les outils de comportement les plus intéressants sont:
Discussions ou scénarios (threads or scenarios), **réseaux de données et flux d'événements**, **théorie des systèmes mathématique**, **théorie du système autonome**, **modèles de comportement et choix du publique**.

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

Discussions ou Scénarios

- ☐ Une discussion ou scénario est une séquence des opérations du système.
- ☐ C'est une liste ordonnée d'évènements et d'actions qui représente un comportement important.
- ☐ Elle ne contient de branches; i.e c'est une série unique de scénario d'opération.
- ☐ Les branches sont représentées par des fils supplémentaires.
- ☐ Les discussions sont importantes pour la communication du client. Pour chaque donnée pose la question « si cette donnée arrive que se passera t il ». Tracer la réponse jusqu'à ce que une output est produite. Et de façon similaire tracer les outputs backward jusqu'à atteindre les inputs.
- ☐ La liste des discussions générées devient une partie des exigences de comportement.

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

Discussions ou Scénarios

- ❑ Les discussions sont aussi importantes pour la communication du constructeur.
- ❑ Fournit des outils intéressants pendant les révisions du design et pour le planning du test.
- ❑ Autre appellation de spécification de comportement par discussions ou scénarios est « **use-cases** ». (cas d'utilisation)
- ❑ Terme plus populaire de spécification de comportement.
- ❑ Ex: le listing d'un dialogue entre le système et un acteur (utilisateur humain du système). L'use case consiste à définir une séquence de messages passés entre le système et l'acteur, augmentés par des explications additionnelles.
- ❑ Use cases sont narratives écrits dans le langage des utilisateurs pour être compris.

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

Réseaux de données et flux d'évènements

- ❑ Un système complexe peut avoir un nbr important de discussions, entraînant une difficulté d'avoir une liste compréhensible.
- ❑ Les réseaux de données et flux d'évènements permet aux discussions d'être groupées en modèles plus compactes mais complets.
- ❑ Les modèles de flux de données définissent le comportement d'un système par un réseau de fonctions ou procès qui échangent les objets de données.
- ❑ Le réseau est généralement défini graphiquement par hiérarchie.
- ❑ Les plus courantes notations utilisent le diagramme de DeMarco (DFD: DeMarco data flow diagram) ou bien diagramme en block du flux fonctionnel (Functional Flow Block Diagram: FFBD).

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

Réseaux de données et flux d'évènements

- ❑ Les 02 méthodes DFD et FFBD sont basées sur un groupe de principes de base:
 1. Les fonctions du système sont décomposées hiérarchiquement. Chaque fonction est composée d'un réseau de sous fonctions jusqu'à obtenir une simple description écrite en texte.
 2. La hiérarchie de décomposition est définie graphiquement.
 3. Les éléments de données sont décomposés hiérarchiquement et sont définis séparément dans un « dictionnaire de données » associé.
 4. Les fonctions sont supposées être des « données déclenchées ».
 5. La structure du modèle évite la définition de redondance. Couplé avec la structuration graphique ça permet au modèle d'être facilement modifié.

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

Théorie des systèmes mathématiques

- ❑ Classiquement, la théorie des systèmes est la théorie du comportement des systèmes de rétroaction (feedback) multidimensionnelle.
- ❑ La théorie de contrôle linéaire est un exemple de théorie de système sur une échelle bien définie limitée.
- ❑ Les formalismes théoriques du système sont construits à partir de 02 composantes:
 1. Une définition de frontière du système en termes de quantités observables, certaines peuvent être soumises au contrôle de l'utilisateur ou du concepteur.
 2. Une machinerie mathématique qui décrit l'évolution dans le temps (le comportement) des quantités de frontière connaissant certaines conditions initiales et stratégies de contrôle.

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

Théorie des systèmes mathématiques

- ❑ Il ya 03 formalismes de systèmes mathématiques principaux qui différent par comment ils traitent les valeurs de données et de temps:
 1. **Systèmes continus:** systèmes modélisés classiquement par des équations différentielles, linéaires et non linéaires. Valeurs sont des quantités continues et sont calculées pour tous les temps.
 2. **Systèmes discrets temporairement (données échantillonnées):** systèmes avec des éléments valorisés continuellement mesurés à des points discrets de temps. Leur comportement est exprimé par des équations de différence. Très importants surtout pour les simulations numériques.
 3. **Systèmes d'évènements discrets:** systèmes où certaines ou toutes les quantités prennent sur des valeurs discrètes en points de temps arbitraires. Les réseaux en chaine « queuing network) est un exemple.

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

Agent autonome, Systèmes Chaotiques

- ❑ Le comportement au niveau système est un comportement qui n'existe pas dans les composantes du système mais qui est obtenu seulement par l'interaction de toutes les composants du système.
- ❑ Récemment on s'intéresse à une classe de système où quelques types de composants qui se multiplient par reproduction interagissent pour créer de nouveaux (émergents) comportements.
- ❑ Ex: les colonies de Fourmis. La programmation du comportement de chaque fourmi individuellement et ses interactions locales chaotiques avec les autres fourmis et l'environnement est suffisant pour des comportements complexes de haut niveau à émerger de la colonie toute entière.
- ❑ Ex: en fabrication: au lieu d'essayer de résoudre le pb d'ordonnancement de travail très complexe, des contrôleurs autonomes programment à travers une interaction distribuée.

4. Classification des modèles par vue (suite)

4.3 Modèles de comportement (fonctionnels) (suite)

Modèles de comportement et choix du publique

- ❑ Certains systèmes dépendent du comportement de la société humaine faisant partie du système.
- ❑ Dans ce cas, les méthodes du choix du publique et l'analyse du consommateur seront nécessaires pour la compréhension du système humain. Ces méthodes sont généralement ad-hoc (improvisé) mais la plupart étaient utilisées en analyse de marketing par les compagnies de produit de consommation.

4. Classification des modèles par vue (suite)

4.4 Modèles de Performance

- ❑ Un modèle de performance décrit ou prédit comment effectivement une architecture satisfait certaines fonctions.
- ❑ Les modèles de performance sont généralement quantitatifs.
- ❑ Ils décrivent les propriétés telles que: sensibilité d'ensemble, précision, latence, temps d'adaptation, poids, cout, fiabilité et d'autres.
- ❑ Les exigences de performance sont souvent appelées exigences « non fonctionnelles » parce qu'elles ne définissent pas une discussion fonctionnelle de l'opération. Ex: le cout n'est pas un comportement du système, mais c'est une propriété importante du système.

4. Classification des modèles par vue (suite)

4.4 Modèles de Performance (suite)

❑ La structure interne des modèles de performance est selon les 03 catégories suivantes:

1. **Analytique**: ce sont les produits des sciences de l'ingénieur. Un modèle de performance de cette catégorie est un groupe de paramètres de niveau inférieur du système et une règle mathématique de la combinaison qui prédit le paramètre de performance d'intérêt.
2. **Simulation**: peut remplacer la règle mathématique de combinaison. C'est un modèle analytique du comportement et de la performance du système en termes de paramètres de simulation.
3. **Par jugement**: où l'analyse et la simulation sont inadéquates ou bien infaisables, le jugement de l'être humain peut conduire à des indicateurs fiables de performance. (les heuristiques).

4. Classification des modèles par vue (suite)

4.4 Modèles de Performance (suite)

Méthodes Formelles

- ❑ En génie du logiciel, ils utilisent les méthodes formelles pour la modélisation de la performance.
- ❑ Ces méthodes ont pour but de développer des systèmes qui produisent de façon formelle les propriétés fonctionnelles et non fonctionnelles.
- ❑ Cette méthode se résume en:
 1. Identifier les inputs et les outputs du système. Identifier un groupe de relations logiques et mathématiques qui doivent exister entre les séquences de l'input et de l'output quand le système est fonctionnel.
 2. décomposer le système en composants, en identifiant les inputs et les outputs de chaque composant. Déterminer les relations mathématiques de chaque composant de façon à ce que leur composition est équivalente au groupe original de relations, un niveau supérieur.

4. Classification des modèles par vue (suite)

4.4 Modèles de Performance (suite)

Méthodes Formelles

- 3. Continuer le procès itérativement au niveau des éléments d'implémentation primitive. En software, ça peut être les déclarations du langage de programmation.**
 - 4. Composer l'implémentation backward, jusqu'à la chaine d'inférence à partir d'éléments primitifs de façon à conserver les relations exactes décomposées. L'implémentation résultante est équivalente à la spécification originale.**
- ❑ Du point de vue Architecte, les plus importantes applications des méthodes formelles sont en phases conceptuelles et en certification de l'assurance élevée et ultraqualité des systèmes.**
 - ❑ Méthodes formelles exigent la détermination explicite des séquences input/output autorisées et non autorisées.**

4. Classification des modèles par vue (suite)

4.5 Modèles de Données

- ❑ Dans les systèmes complexes, il ya pb de **retenue de données**.
- ❑ Quelles sont les données le système retient et quelles sont les relations entre les données le système développe et maintient?
- ❑ Les modèles de données les plus utilisés sont issus du développement de logiciels surtout développement de grandes bases de données.
- ❑ Ces méthodes ont été développées pour répondre aux besoins des systèmes **data-intensive automatisé** et les systèmes **paper-based**.
- ❑ Les systèmes de données intensives sont les grands systèmes de bases de données automatisé.
- ❑ Systèmes paper-based permet de capturer les interrelations complexes entre de grandes quantités de données retenus.
- ❑ Ces méthodes de données sont de plus en plus utilisées vue la tendance actuelle d'introduire beaucoup d'intelligence dans les systèmes.

4. Classification des modèles par vue (suite)

4.5 Modèles de Données (suite)

- ❑ La base des modèles de données modernes sont les diagrammes développés pour les bases de données relationnelles.
- ❑ Ces diagrammes ont été généralisés en famille de techniques de modélisation en orienté objet.
- ❑ Les principales types de relations incluent la composition, la généralisation, la spécialisation, héritage et association.
- ❑ Méthodes de modélisation orienté objet combine la modélisation du comportement et des données en une seule hiérarchie organisée le long des données.
- ❑ Un modèle orienté objet commence par la décomposition des données et lui accroche un modèle fonctionnel.
- ❑ Décompositions données orientées partagent les heuristiques générales de l'architecture du système.

4. Classification des modèles par vue (suite)

4.6 Modèles Managériaux

- ❑ Pour l'architecte et le client un projet est une partie technique, une partie management (planning, budgets, schedule...)
- ❑ En systèmes socio économiques, planning le déploiement du système est plus difficile que l'assemblage de son hardware.
- ❑ La vision managériale ou implémentation décrit le procès de construire le système physique.
- ❑ Aussi, il suit les événements de construction qd ils se produisent.
- ❑ L'architecte a 02 intérêts principaux dans les modèles managériaux:
 1. Le client ne peut pas avancer dans son projet sans avoir des estimations solides de couts et de délais. Ces estimations nécessitent de gros efforts dans les modèles de management.
 2. Le client peut être appelé pour suivre le système lors de son construction pour s'assurer de l'intégrité de la conception.

5. Exemples de Modèles Intégrés.

- ❑ Les modèles qui **intègrent plusieurs vues** sont ceux qui concernent le plus l'architecte.
- ❑ Ces modèles intégrés fournissent la **vue synthétisée centrale** aux préoccupations de l'architecte.
- ❑ Une méthode de modélisation intégrée est une représentation du système qui relie plusieurs vues.
- ❑ La méthode consiste en un groupe de modèles pour un sous-groupe de vues et un groupe de règles ou bien modèles additionnels pour relier les vues fondamentales.
- ❑ La plupart des méthodes de modélisation intégrée s'appliquent à un domaine particulier (tableau suivant).

5. Exemples de Modèles Intégrés. (suite)

Méthode	Domaine
Hatley/Pirbhai (H/P)	Systèmes réactifs basés sur le computer.
Quantitative quality function deployment (Q ² FD)	Systèmes avec des objectifs de performance quantitatif important et les modèles de performance compris.
Techniquement de modélisation objet (OMT)	Systèmes software data-intensive, grande échelle, surtout ceux implémentés en langages objet modernes
ADARTS (Ada-based Design Approach for Real-Time Systems)	Systèmes software real-time, grande échelle.
Analyse de systèmes de fabrication (MSA)	Systèmes de fabrications intelligents.

5. Exemples de Modèles Intégrés. (suite)

Comparaison des méthodes de modélisation les plus populaires

Vue	H/P	OMT	ADARTS	Q²FD	MSA
Objectifs	Text	Text	Text	Nombres	Text
Comportement	Flux de control/données	Diagram. De classe, flux de données, diagrm. D'état	Flux d'événement/données	Liens seulement	Flux de données
Performance	Text (seulement text de timing)	text	text	Modèles de satisfaction, matrices QFD	Text, liens aux modèles d'ordonnancement standads
Données	dictionnaire	Diagrammes objet/classe	Dictionnaire	N/A	Diagram. Entité-relation
Forme	Diagram. En blocks	Diagram. objets	Diagram. Structure tache-objet	Liens par allocation	Diagram. De flux procés ASME
Managérial	N/A (liens à travers des mesures)	N/A	N/A (liens à travers des mesures)	N/A	Modèle de flux de fonds, comportement d'ordonnancement

6. Résumé

- ❑ Le travail d'un architecte tourne autour de modèles.
- ❑ Puisque l'architecte ne construit pas directement le système, son intégrité pendant la construction doit être maintenue à travers des modèles agissant comme **substituts**
- ❑ Les modèles représentent et contrôlent la spécification du système, son design et son plan de production.
- ❑ Même après la livraison du système, la modélisation sera le mécanisme pour évaluer le comportement du système et planifier son évolution.
- ❑ La modélisation pour les architectes est conduite par 03 caractéristiques clés:
 1. Les modèles sont les langages principaux de l'architecte. Leur rôle est de faciliter la communication avec le client et le constructeur. Avec la communication ils assument aussi l'intégrité du design et la synthèse.

6. Résumé (suite)

- 2. Les architectes exigent une multiplicité de vues et de modèles. Les plus basics sont objectif, forme, comportement, performance, données et management.**
- 3. La multidisciplinarité, méthodes de modélisation intégrée lie ensemble les différentes vues. Elles permettent à raffinement du design en étapes de l'abstrait conceptuel au détail nécessaire à la construction du système.**

Systems Architecting

Abdellatif MEGNOUNIF

Semaine Prochaine

**Progression du Design en
architecting des systèmes**

Merci. Fin du chapitre 9