

Systems Architecting

Abdellatif MEGNOUNIF

Chap. 3

Les Heuristiques

COURS 3 Jeudi 03.11.2011

© **Abdellatif MEGNOUNIF** FT-Tlemcen



1. Introduction

- ❑ Il ya 02 types de personnes: Ceux qui savent compter et d'autres qui ne savent pas.
- ❑ Ceux qui savent compter (les ingénieurs) approchent leurs pbs de design en utilisant l'analyse et l'optimisation, des outils puissants et précis dérivés de méthodes scientifiques et de calculs.
- ❑ Ceux qui ne savent pas (architectes) approchent leurs pbs qualitatifs en utilisant des directives, des abstractions et des pragmatiques générées par des leçons apprises de l'expérience (heuristiques).
- ❑ Les outils utilisés sont différents puisque les pbs à résoudre sont différents.
- ❑ Pour les architectes il ya des centaines d'outils, mais pour une période définie et pour un travail spécifique on utilise qlq outils.

1. Introduction (suite)

- ❑ Pour continuer le métaphore, le peu d'outils que l'architecte utilise, il est préférable qu'il utilise des outils d'autres (supply store) que ça soit pour le hardware, le software ou les heuristiques.
- ❑ (A la fin de ce chapitre on présentera une liste d'heuristiques pour l'architecting des systèmes.
- ❑ Les clients parcourent d'abord puis sélectionnent un kit d'outils basé sur le travail, la compétence personnelle et la connaissance de l'origine et de l'utilisation attendue de chaque outil.

Métaphore: procédé par lequel on transporte la signification propre d'un mot à une autre signification qui ne lui convient qu'en vertu d'une analogie, d'une comparaison sous entendue. (la fleur de l'âge, la lumière de l'esprit)



2. Définition

- ❑ **Heuristique:** mot grec, heuriskein, qui veut dire « pour trouver un chemin » ou bien « pour guider » (dans le sens piloter un bateau à travers un fond dangereux)
- ❑ **Larousse:** trouver. Qui sert à la découverte, notamment dans la recherche scientifique et épistémologique
- ❑ **Un bref énoncé de la vision d'une situation, une leçon apprise, ou une directive du design. Ce sont les produits d'un raisonnement inductif, un processus de l'estimation réelle face à une connaissance incomplète, qui mélange l'information connue par expérience avec une conjecture plausible. (Klir 1985).**
- ❑ **Exemples:** i) quand vous divisez le système choisir les éléments qui ont une complexité interne élevée et une complexité externe basse (cohésion élevée et couplage bas)
ii) Toutes les erreurs réellement importantes sont faites au début.

2. Définition (suite)

- ❑ **Architecting est aussi une forme de pilotage de bateau.**
 - ❖ **Ses rochers et bas-fonds sont les risques, les changements technologiques, l'environnement opérationnel et de construction qui caractérise les systèmes complexes.**
 - ❖ **Son port sécurisé sont l'acceptation et la sécurité du client, vie longue et dépendante.**
- ❑ **Les heuristiques sont des guides le long du chemin qui doivent être utilisées avec jugement. (02 ports ne se ressemblent pas)**
- ❑ **Les guides peuvent ne pas garantir un passage sécurisé mais les oublier ça peut être fatal.**
- ❑ **Les enjeux en architecting sont élevés comme réputation, ressources, services vitaux et vies.**

2. Définition (suite)

Une bonne heuristique:

- ❖ **Doit refléter un large consensus qu'il est fortement corrélée avec les résultats dans leur contexte original**
- ❖ **Doit être validée, mais ne pas être démontrable**
- ❖ **Peut se produire comme partie d'un processus, chemin ou progression**

Le but de ce chapitre est d'aider les étudiants (architecte, ingénieur ou manager) à trouver et développer des heuristiques qui peuvent être prises avec confiance, les organiser selon les besoins et les utiliser en pratique.

3. Heuristiques comme abstractions de l'expérience

- ❑ L'une des caractéristiques de l'être humain est non seulement sa capacité d'apprendre mais d'aller vers le futur (plus développé) avec des leçons apprises de l'expérience.
- ❑ Chaque génération connaît plus, apprend plus, planifie plus, essaye plus et réussissent plus que celle qui la devance parce qu'elle profite de leur expérience (gain de temps).
- ❑ Pas beaucoup d'architectes travaillent sur 02 ou 03 systèmes complexes dans leur vies (trop de temps). Ils n'ont ni le temps ni l'opportunité d'avoir de l'expérience voulue pour la création de systèmes complexes à partir de zéro.
- ❑ Par le même processus, des heuristiques qualitatives, des expériences pratiques codifiées et condensées apparaissent pour compléter les équations et les algorithmes de la science et de l'ingénierie pour résoudre les pbs complexes.
- ❑ Passé d'un architecte à un autre d'un système à un autre ils travaillent et aident à satisfaire un vrai besoin.

Abstraction: action d'abstraire (ne pas en tenir compte)



3. Heuristiques comme abstractions de l'expérience (suite)

- ☐ La forme des heuristiques (en contraste avec les symboles de physique et maths) sont des mots exprimés en langages naturels.
- ☐ Elles traduisent les cultures d'ingénierie, d'exploration et des relations humaines.
- ☐ La naissance d'une heuristique commence par des anecdotes et des histoires qui deviennent des fables et des mythes (c'est un processus d'un raisonnement inductif).
- ☐ Leurs impact est non seulement sur les politiques, religion et business mais aussi sur le design technique de systèmes et services.
- ☐ Les leçons sont celles appliquées au-delà du contexte original, étendue par analogie, par comparaison, par conjoncture et par testing.
- ☐ Elles sont vues comme des réalités self-évidentes ne nécessitant pas de démonstration.



4. Sélection d'un kit personnel d'outils en heuristique

« The art in architecting lies not in the wisdom of the heuristics, but in the wisdom of knowing which heuristics apply, a priori, to the current project ».

Williams, P. L., 1992 Systems Architecting Report, University of Southern California.

Abstraction: action d'abstraire (ne pas en tenir compte)

4. Sélection d'un kit personnel d'outils en heuristique (suite)

- ❑ Toutes les professions et leurs praticiens ont leurs propres kit d'outils (physique ou heuristique) sélectionnés de leurs expériences et celle des autres en fonction de leurs besoins et leurs talents.
- ❑ Qlq chercheurs ont eu l'idée de créer une sélection plus large en collectant ensemble des leçons apprises de domaines différents (Aerospace, électronique, informatique...) en les exprimant sous forme heuristique pour les utiliser par les architectes, les éducateurs, les chercheurs et les étudiants.
- ❑ Une collection initiale de 100 heuristiques a été dépassée rapidement pour atteindre 1000 en 06 ans.
- ❑ La plupart sont des variations d'idées centrales et individuelles répétées plusieurs fois dans des contextes différents.
- ❑ Les 04 heuristiques les plus utilisées sont, dans l'ordre décroissant de popularité:

4. Sélection d'un kit personnel d'outils en heuristique (suite)

1. **D'ont assume that the original statement of the problem is necessarily the best, or even the right one.**

Exemple: la déclaration initiale du pb de l'avion de combat F-16 demandé pour une capacité supersonique élevée qui était difficile et très chère à avoir.

Discussions avec l'architecte (Harry Hillaker) fait sortir que la raison pour cette déclaration était de donner « une sortie rapide du combat » beaucoup mieux que le rapport poussée/poids, supersonique bas

En qlq sorte, la déclaration d'une vitesse élevée a été remplacée par celle d'une accélération élevée avec l'ajout de l'avantage d'une manœuvrabilité exceptionnelle.

4. Sélection d'un kit personnel d'outils en heuristique (suite)

2. **In partitioning, choose the elements so that they are as independent as possible; that is, elements with low external complexity and high internal complexity.**

Exemple: un des pbs difficile dans le design des puces électroniques est l'utilisation efficace de leur surface dont la majorité est consommée par les connections entre les composants (i.e par communication au lieu du processing)

Pr Carver Mead a démontré que le design basé sur le minimum de communication entre composants donne plus d'utilisation efficace de la surface et la puce apparaisse plus élégante. (un bon signe d'une architecture fine qui confirme une autre heuristique **the eye is a fine architect. Believe it.)**

4. Sélection d'un kit personnel d'outils en heuristique (suite)

3. **Simplify. Simplify. Simplify.**

Exemple: l'une des meilleures techniques pour augmenter la fiabilité en diminuant le prix et le temps est de réduire le nbr de pièces d'un dispositif.

Les ingénieurs de l'automobile ont récemment eu de résultats remarquables en substituant la coulée unique par des assemblages multiples et en réduisant le nbr d'attaches et la difficulté de l'assemblage associé par de meilleurs placements.

4. Sélection d'un kit personnel d'outils en heuristique (suite)

4. **Build in and maintain options as long as possible in the design and implementation of complex systems. You will need them.**

**Exemple: en industrie d'avions on les appelle « scars » (cicatrices).
En logiciels on les appelle « hooks » (hameçon).**

Les 02 sont des interruptions planifiées ou des points d'entrée dans le système qui peut étendre les fonctions du système.

Dans les avions, elles sont utilisées pour allonger plus le fuselage pour prendre plus de passagers ou charges.

Pour le software, elles sont utilisées pour insérer plus de routines.

4. Sélection d'un kit personnel d'outils en heuristique (suite)

- ☐ Bien que ces 04 heuristiques ne font pas un kit complet, elles sont un bon exemple pour construire un kit.
- ☐ Toutes ont un seul objectif (principal en architecting des systèmes) **réduire la complexité**.
- ☐ Toutes ont été utilisées par confiance sous une forme ou autre et dans plusieurs domaines.
- ☐ Toutes ont vérifiées le test du temps.

4. Sélection d'un kit personnel d'outils en heuristique (suite)

- ❑ La 1^{ère} étape dans la création d'un kit important d'heuristiques est de déterminer les critères de sélection, dont certains sont comme suit:
 - ❖ L'heuristique doit avoir un sens dans son domaine ou contexte original. Pour être acceptée, une forte corrélation (sinon cause-effet) doit apparaître entre l'heuristique et les succès ou échecs de systèmes spécifiques, produits ou procès.
 - ❖ Le sens général, sinon les mots spécifiques, de l'heuristique devrait s'appliquer au-delà du contexte original. i.e qu'elle doit résoudre ou expliquer plus que le pb original. (ex: **before the flight it's opinion; after the flight it's obvious**). Cette heuristique s'applique pour les systèmes ultra qualité.
 - ❖ L'heuristique doit être facilement rationalisée en qlq minutes ou bien moins d'une page. (ex: **if you can't explain it in five minutes, either you don't understand it or it doesn't work**). Le plus évident l'heuristique est sur sa face, et moins les limitations en utilisation c'est mieux (**a model is not reality**)



4. Sélection d'un kit personnel d'outils en heuristique (suite)

- ❖ La déclaration opposée d'une heuristique doit être insensée, clairement « non sens commun ». Ex: l'opposée de « **if it can fail, it will** (loi de Murphy) peut être « **if it can fail, it will not** » (non sens).
- ❖ la leçon de l'heuristique doit résister au facteur temps et gagner un consensus plus large.

4. Sélection d'un kit personnel d'outils en heuristique (suite)

- ☐ Les heuristiques peuvent aussi être utilisées en groupe.
- ☐ Aussi une action ou décision peut être plus forte si elle est liée à plusieurs heuristiques au lieu d'une seule. Un groupe d'heuristiques appliqué aux procédures d'acceptation justifie cette proposition.
- ☐ Une heuristique prise dans un contexte suffisamment restreint peut être spécialisée dans une règle de design, une évaluation rationnelle quantifiable ou bien un algorithme de décision.

Dans ce cas les heuristiques sont des ponts utiles entre architecting, engineering et design.

5. Utilisation des heuristiques

- ❑ **Tout le monde sait que les heuristiques jouent un rôle important dans le design et développement des activités.**
- ❑ **Si on sait que beaucoup utilisent les heuristiques, il n'est pas évident que ces heuristiques sont communiquées et utilisées par d'autres.**
- ❑ **Les heuristiques peuvent être transférées de quelqu'un à un autre mais pas nécessairement de façon simple.**
- ❑ **Les gens souvent utilisent les heuristiques sous 03 formes:**
 - 1. Peuvent être utilisées comme guides évocateurs (qui font penser à de nouvelles pensées chez le lecteur). Certains utilisent un catalogue des heuristiques qd ils ont un pb de design. Si une des heuristiques paraisse intéressante, ils l'utilisent.**

5. Utilisation des heuristiques (suite)

- 2.** **Peuvent être utilisées comme codification de l'expérience.**
L'heuristique est comme un aperçu, un guide à une discussion détaillée qui suivra. Dans ce cas les histoires derrière l'heuristique sont plus importantes que la déclaration de base.
L'heuristique est un outil pédagogique.

- 3.** **Peuvent être intégrées dans des processus de développement (très intéressant).**

Exemple du software: la programmation orientée objet.
Commence par des exigences textuelles qui seront transformés en classes et objets (de façon non unique). Ce qui provoque des jugements extensifs par le praticien. Certaines méthodes fournissent de l'assistance au praticien en donnant des heuristiques prescriptives et explicites pour chaque étape.

6. Heuristiques sur heuristiques

- ❑ Keneth L. Cureton génère un groupe d'heuristiques sur comment générer et appliquer les heuristiques, comme:

6.1 Génération d'heuristiques utiles.

- ❑ L'humour en heuristique fournit un mordant émotionnel qui renforce l'effet mnémonique (relatif à la mémoire)
- ❑ Utiliser des mots qui transmettent une frisson de perspicacité (thrill of insight) dans l'esprit du spectateur.
- ❑ Pour un effet maximal, essayer d'intégrer des messages descriptifs et prescriptifs dans les heuristiques.
- ❑ Plusieurs heuristiques peuvent être employées à des heuristiques elles-mêmes. Ex: **Simplify. Scope**
- ❑ Ne rend pas l'heuristique tellement élégante qu'elle n'aura que le sens de son créateur, perdant ainsi le principe de l'utilisation générale.
- ❑ Au lieu d'ajouter une déclaration à l'heuristique, créer une autre séparées et associée qui répond à cette situation additionnelle.

6. Heuristiques sur heuristiques (suite)

6.2 Application des heuristiques

- ☐ Si ça marche, alors c'est utile.
- ☐ Connaissant quand et comment utiliser une heuristique est aussi important que connaître les quoi et pourquoi (what and why).
- ☐ Les heuristiques fonctionnent mieux quand elles sont appliquées tôt pour réduire l'espace de la solution.
- ☐ Viser l'équilibre. « Beaucoup de bonne chose ou bien éliminer complètement les mauvaises choses peuvent rendre les choses mauvaises (pas bonnes) »
- ☐ Pratiquer. Pratiquer. Pratiquer.
- ☐ Heuristiques ne sont pas la réalité.

7. La taxonomie des heuristiques

- ❑ Après avoir trouver ou créer des heuristiques individuelles, il faut les organiser pour une utilisation facile de façon à ce que les plus appropriées sont à portée de la main pour une activité donnée.
- ❑ La collection peut se faire selon les taches architecturales.
- ❑ Peut aussi se faire selon les phases du modèle en cascade.
- ❑ Les taches peuvent être:
 - ✓ Scoping et planning (délimitation et planification)
 - ✓ Modélisation
 - ✓ Priorisation
 - ✓ Agrégation
 - ✓ Partitionnement
 - ✓ Intégration
 - ✓ Certification
 - ✓ Evaluation
 - ✓ Evolution et rearchitecting

8. Résumé

- ❑ Les heuristiques, abstraction d'expérience, sont des guides non analytiques, dignes de confiance pour traiter des pbs complexes, mal structurés et intrinsèquement sans limité.
- ❑ Utilisés en aide de prise de décision, jugements de valeur et évaluations.
- ❑ Trouvées à travers l'architecting des systèmes, de la conceptualisation tôt au diagnostique et opération.
- ❑ Constituent des ponts entre client et constructeur, concept et implémentation, synthèse et analyse, système et sou système.
- ❑ Sont des transitions successives des besoins qualitatives et provisoires aux directives prescriptives et descriptives et donc aux approches et méthodes rationnelles.
- ❑ Comment trouver, créer, organiser et utiliser les heuristiques pour traiter les pbs qualitatifs de l'architecting des systèmes.

9. Appendix (heuristiques pour l'architecting des systèmes)

Experience is the hardest kind of teacher. It gives you the test and the lesson afterward. (Susan Ruth 1993)

- ☐ Performance, cout et délai ne peuvent pas être spécifiés indépendamment. Au moins un des trois doit dépendre des 02 autres. (descriptif)
- ☐ Pour peu d'exceptions, les retards sont acceptés à contre cœur; les surcouts ne le sont pas, et pour une bonne raison. (descriptif)
- ☐ Le temps de compléter est proportionnel au rapport du temps dépensé sur le temps planifié à ce jour. Plus le rapport est grand plus le temps qui reste. (descriptif)
- ☐ Les relations entre les éléments sont ce qui donne aux systèmes leur valeur ajoutée. (descriptif)
- ☐ L'efficacité est inversement proportionnelle à l'universalité. (descriptif)

9. Appendix (heuristiques pour l'architecting des systèmes) (suite)

- ☐ Si quelque chose peut tourner mal, tournera mal (Loi de Murphy).
- ☐ Simplifier. Simplifier. Simplifier. (prescriptif)
- ☐ Simplifier, combiner et éliminer. (prescriptif)
- ☐ Simplifier avec des éléments intelligents. (prescriptif)
- ☐ L'élément le plus fiable dans un avion c'est celui qui n'est pas là (parce qu'il n'est pas utile). (prescriptif)
- ☐ Une architecture d'une personne est le détail d'une autre personne. Un système d'une personne est un composant d'une autre.
- ☐ Dans le but de comprendre quoi que ce soit, vous ne devez pas essayer de comprendre tout (aristote, 4th siècle BC)
- ☐ Ne pas confondre le fonctionnement d'une composante et le fonctionnement du système.
- ☐ En général, chaque niveau du système fournit un contexte pour le(s) niveau(x) inférieurs.

9. Appendix (heuristiques pour l'architecting des systèmes) (suite)

- ☐ Laisser les spécialités aux spécialistes. Le niveau de détail exigé par l'architecte est seulement à la profondeur de l'élément ou la composante critique au système comme un tout. Mais l'architecte doit avoir accès à ce niveau et connaître (ou bien être informé) sur sa criticité et son statuts.
- ☐ En introduisant des changements technologiques et sociaux, comment vous le faites est plus important que ce que vous faites.
- ☐ Si la politique ne s'envole pas le hardware ne le sera jamais.
- ☐ Politique, non la technologie, définit les limites de ce que devrait faire la technologie.
- ☐ The best engineering solutions are not necessary the best political solutions.
- ☐ Pour rester compétitif, déterminer et contrôler les clés de l'architecture du début.

9. Appendix (heuristiques pour l'architecting des systèmes) (suite)

- ☐ The beginning is the most important part of the work (plato, 4th cen BC)
- ☐ Le succès est défini par le spectateur et non pas par l'architecte.
- ☐ le plus important élément de succès est d'écouter attentivement à ce que le client perçoit comme ses exigences et avoir la volonté et la possibilité de répondre.
- ☐ Demander très tôt sur comment vous allez évaluer le succès de vos efforts.
- ☐ Les points forts d'un système ou d'une organisation dans un contexte peuvent être des faiblesses d'un autre. Connaitre quand et où.
- ☐ Un système est réussi c'est l'intersection naturelle de la technologie, du politique et de l'économie est trouvée.

9. Appendix (heuristiques pour l'architecting des systèmes) (suite)

- ☐ **Aucun système complexe peut être optimal à toutes les parties concernées, ni toutes les fonctions optimisées.**
- ☐ **Dans certains cas, il est important de connaître qui est le client que ce qu'il veut.**
- ☐ **Dans certains cas, pas toujours, le meilleur moyen de résoudre un pb est de l'élargir.**
- ☐ **Ne rend pas l'architecture plus intelligente pour son propre bien.**
- ☐ **Don't assume that the original statement of the problem is necessary the best, or even the right, one.**
- ☐ **Work forward and backward.**
- ☐ **L'ordre dans lequel les décisions sont prises peut changer l'architecture aussi bien que les décisions elles-mêmes.**
- ☐ **Les réalités à la fin de la phase conceptuelle ne sont pas les modèles mais les critères d'acceptation.**

9. Appendix (heuristiques pour l'architecting des systèmes) (suite)

- ☐ If you can't analyze it, don't built it.
- ☐ La modélisation est un métier et parfois un art.
- ☐ Une vision est une architecture imaginaire...pas mieux, pas mauvais que le reste des modèles.
- ☐ Un modèle n'est pas une réalité.
- ☐ Une clairvoyance (insight) équivaut à 1000 analyses.
- ☐ Si vous ne pouvez pas l'expliquer en 5 mn, soit vous ne l'avez pas compris soit ça ne fonctionne pas.
- ☐ L'œil est un architecte fin

Systems Architecting

Abdellatif MEGNOUNIF

Semaine Prochaine

Partie II: Nouveaux domaines, nouvelles idées

Systèmes avec architecture de constructeur

Merci. Fin du chapitre 3