Ingénierie des exigences basée sur les buts

Michael Petit (FUNDP)
Christophe Ponsard (CETIC)

Février 2003

Plan

- Présentation du cas utilisé
- Motivations de l'approche par les buts
- Application en parallèle de 2 représentants typiques: i* et KAOS
  - modèle idéal
  - modèle plus réaliste
  - spécificités des approches
  - outils
- Synthèse et discussion

Exercice

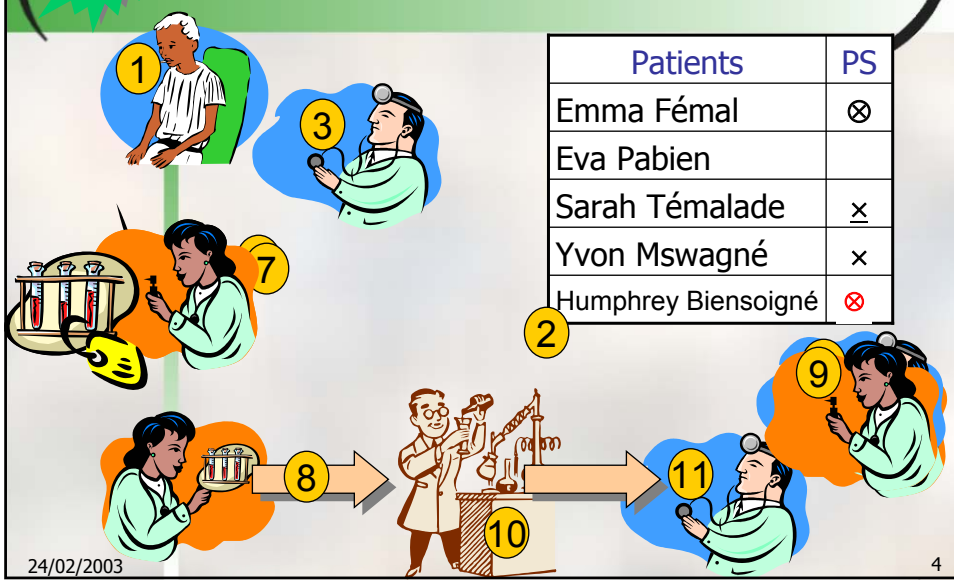
Le cas étudié

Informatisation des consignes de prises de sang dans un Service d'Urgences hospitalier

Exercice

Service d'Urgences

Patients	PS
Emma Fémal	⊗
Eva Pabien	
Sarah Témalade	×
Yvon Mswagné	×
Humphrey Biensoigné	⊗



Qu'est-ce l'Ingénierie des Exigences ?

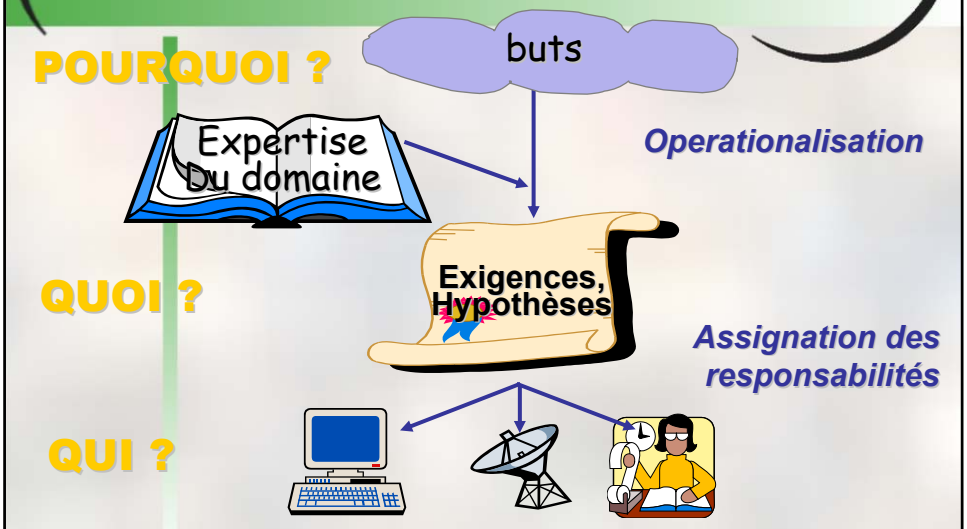
Ross'77

- Les exigences doivent énoncer:
  - **pourquoi** un système est nécessaire, sur base des conditions actuelles ou prévues
  - **quelles caractéristiques** du système y contribueront
  - **la manière** dont le système doit être construit

Zave'97

- L'IE se préoccupe des **buts**, **fonctions** et **contraintes** réelles de systèmes logiciels; et de leurs liens avec:
  - les spécifications précises du comportement logiciel
  - leur évolution au cours du temps

Qu'est-ce l'Ingénierie des Exigences ?



Méthodes d'analyse et leurs limites

Exemples:

- SADT
- Merise
- OMT/**UML**
- **…**

Mais

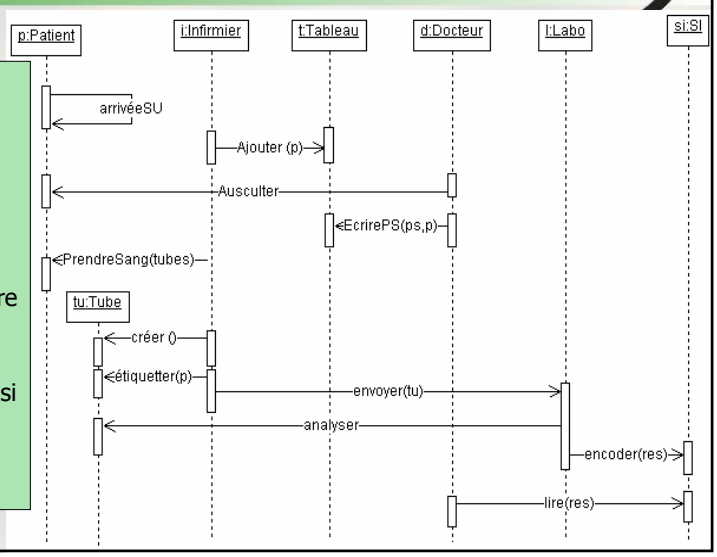
- limités aux aspects **fonctionnels**
- souvent exprimés de manière trop **opérationnelle**
- faiblesse de l'aspect **contractuel**

Bon pour des spécifications de logiciels
Mais insuffisant pour les cahiers de charges

Exemple de limitation: diagramme de séquence UML

Aspects non explicités:

- **Pourquoi** colle-t-on des étiquettes sur les tubes?
- Y a-t-il un **autre moyen** de faire des analyses (dans un autre hôpital, …)? Si oui, quels avantages?
- Est-ce **satisfaisant** si les résultats sont envoyés sur papier?
- …



Buts : définition

« Objectifs à satisfaire par le système »

- Propriétés intensionnelles
- Propriétés requises (prescriptives) ou souhaitées (optatives)
- >< propriétés descriptives du domaine

- Système:
  - logiciel + environnement
  - système existant vs système en conception

Les buts: différents niveaux d'abstraction

- Buts de haut niveau
  - stratégiques, globaux, couvrant l'organisation
  - "assurer la prise en charge des patients"
  - "assurer de bonnes conditions de travail au personnel hospitalier"
- Buts de bas niveau
  - opérationnels, locaux, spécifiques au design
  - "des étiquettes portant le nom du patient doivent être apposée sur les tubes"
  - "l'infirmier doit signaler la prise en charge du patient sur le tableau"

Buts fonctionnels (FR) et non-fonctionnels (NFR)

- Buts fonctionnels: **ce que le système doit faire**
  "Prise en charge du patient"
- Buts non-fonctionnels:
  qualités supplémentaires souhaitées
  - « Safety » : "pas d'erreur médicales"
  - « Security » : "confidentialité du dossier médical"
  - Performance : "traitement rapide"
  - Coût : "rentabilité du service"
  - Adaptabilité : "déclenchement d'un plan d'urgence"

Réalisation coopérative des buts

- coopération d'agents: logiciels, matériels et/ou humains:
  eg. prise de sang: médecin, patient, labo, SI
- agent responsable d'un but
- assignation d'un but
  - dans le système = exigence
    - eg. apposer les étiquettes
  - dans l'environnement = hypothèse, norme, politique
    - eg. patient coopératif

Apports des buts

Nous allons voir que les buts…

- dirigent l'élaboration des exigences et permettent de les motiver (critère de pertinence)
  - approche « top-down » (HOW?): nouveau système
  - approche « bottom-up » (WHY?): évolution de l'existant
- fournissent de riches mécanismes de structuration facilitant la compréhension du CdC
- permettent l'examen d'alternatives dans les raffinements et d'assignations de responsabilités avec diverses frontières entre le système et son environnement

Différents modes de représentation

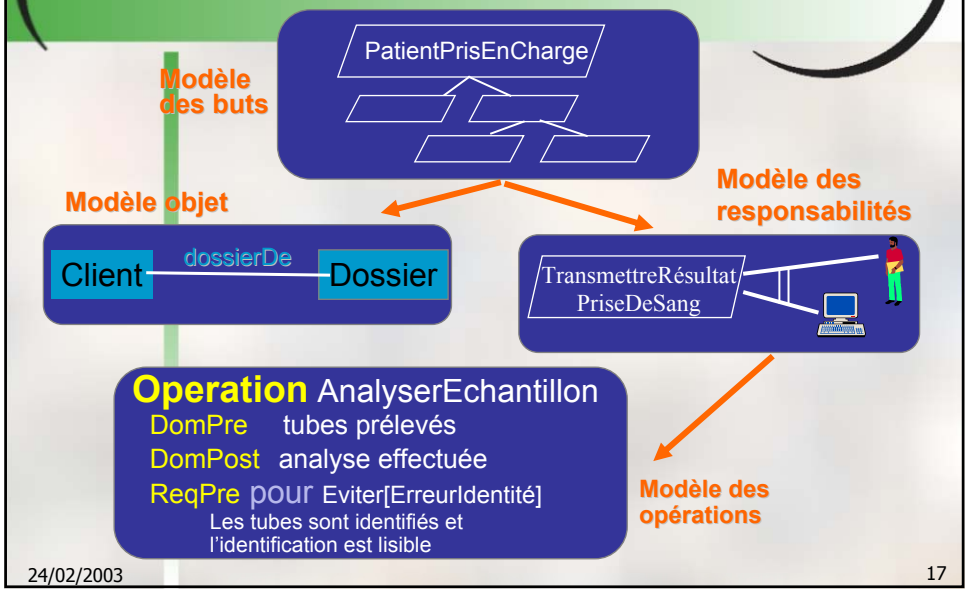
- Informelle: texte (structuré)
  - Canevas de CdC (eg. IEEE-Std-830)
- Semi-formelle: construction d'un modèle
  - **i* [Yu, Mylopoulos et al]**
  - **KAOS [van Lamsweerde et al]**
  - CSD [Feather]
  - Inquiry Cycle [Potts, Anton]
  - EKD [Bubenko, Rolland, Loucopoulos]
  - Win-Win [Boehm]
  - NFR [Chung, Mylopoulos]
- Formelle: utilisation de notations mathématiques. Apport supplémentaire: raisonnement

Une utilisation de i* et de KAOS

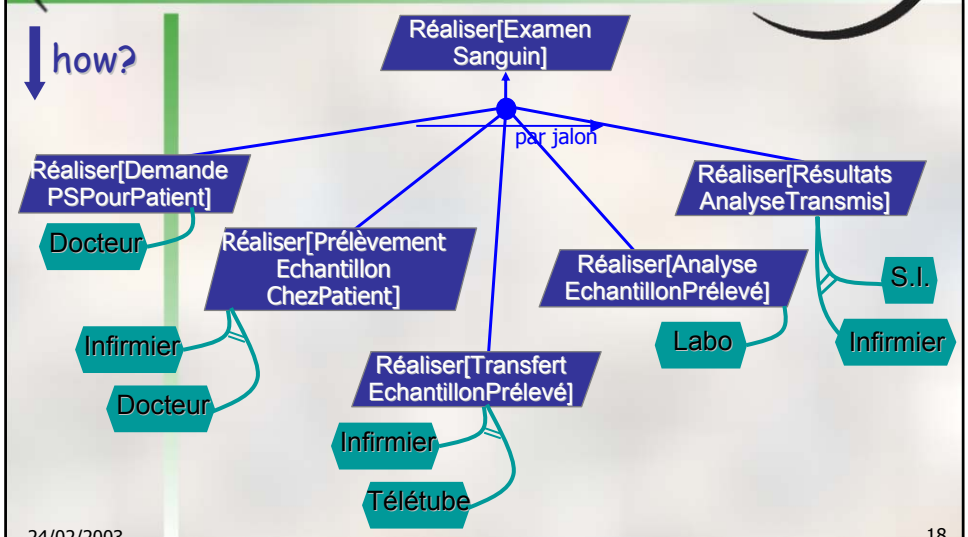
- Poser un diagnostic sur une situation **existante**:
  - Etudier le processus existant
  - Identifier les objectifs (FR et NFR)
  - Raffiner et assigner
  - Evaluer ces objectifs (labellisation i*)
- Déduire des objectifs pour le système informatique

Identification préliminaire des objectifs

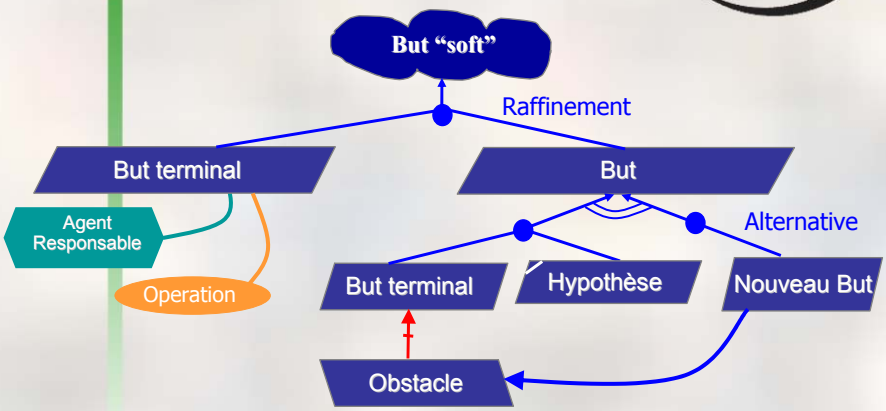
Introduction à KAOS



Raffinement du but fonctionnel



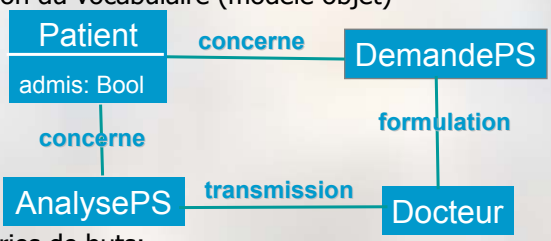
Syntaxe graphique du modèle des buts de KAOS



Détail d'un but

Réaliser[Examen Sanguin]

- Définition informelle:
  Toute demande d'examen sanguin d'un patient admis au service des urgences est traitée dans un délai raisonnable et le résultat de l'analyse du patient concerné est transmis au médecin ayant formulé la demande.
- Elicitation du vocabulaire (modèle objet)

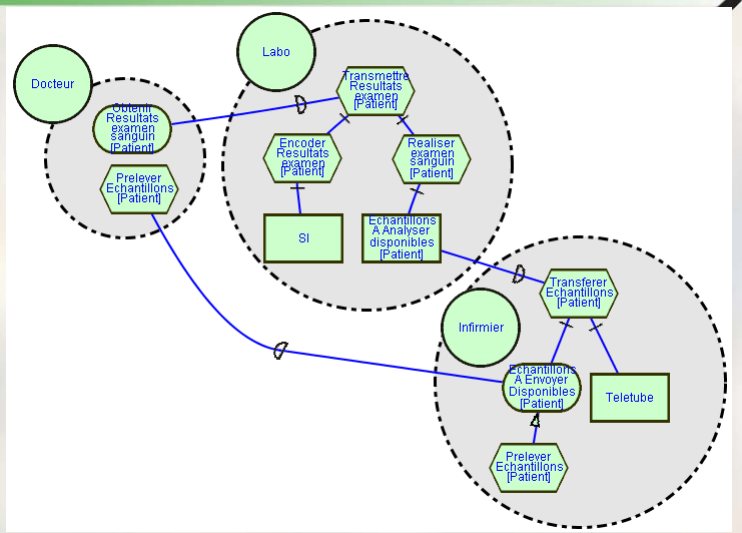


- Catégories de buts:
  - réaliser → un jour
  - assurer → toujours
  - éviter → jamais

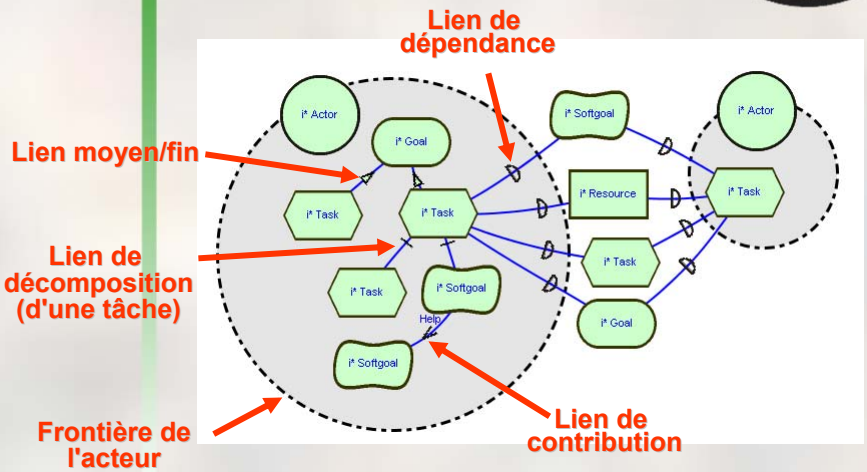
Introduction à i*

- Objectif:
  - Représenter les aspects intentionnels (motivations, intentions, raisonnement)
  - Représenter des configurations organisationnelles (acteurs et dépendances)
  - Vue stratégique: implications à long terme, opportunités, vulnérabilité
- i* = intentionnalité distribuée
  - Acteurs avec buts et capacités
  - Buts individuels ou partagés
  - Buts éventuellement conflictuels
- Deux modèles:
  - Raisonnements stratégiques
  - Dépendances stratégiques

Exemple: Réaliser un examen sanguin



Syntaxe i*: Raisonnement Stratégique



Concepts i*/KAOS: Goal

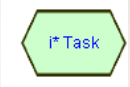
KAOS goal

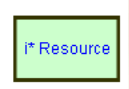
- Goal = But
- Descriptibles et vérifiable de manière précise et objective
- Presque toujours fonctionnel
- Paramètres possibles

Concepts i*/KAOS: Softgoal

- Softgoal= but soft
- Descriptibles de manière imprécise ou subjective. Evaluation possible de la contribution.
- Souvent de nature qualitative (non-fonctionnel)
  - Ex: bien-être du patient, rapidité, convivialité, maintenabilité, …
- Catégories [Chung et al. 1999]
  - Security, Efficiency, Accuracy, …

Concepts i*: tâche et ressource



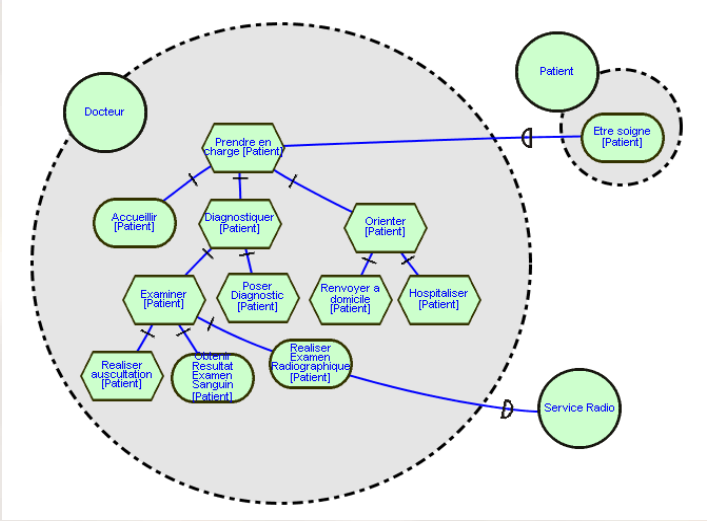


- Tâche (Task): "Procédure" mise en place par l'agent pour atteindre un objectif (pas nécessairement complète)
- Ressource: possédée ou requise par l'acteur (souvent dans le cadre d'une tâche)

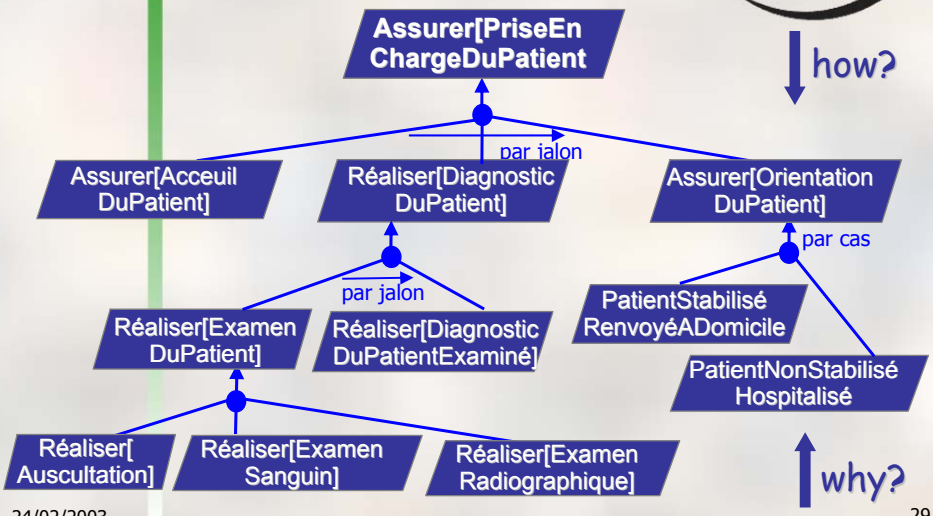
Concepts i*: Liens

- **Décomposition** de tâche: exprime les éléments de la procédure (sous-objectifs, sous-tâches, ressources ou critères de qualité)
- Lien **moyen-fin**: montre l'utilité d'un nœud pour un autre (tâche pour atteindre un objectif, objectif pour un autre, …)
- Lien de **contribution**: montre l'influence (positive ou négative) d'un nœud sur un autre. Types de liens différents en fonction du degré d'influence
- Lien de **dépendance**: acteur se repose sur un autre pour un objectif, une tâche, un ressource, …

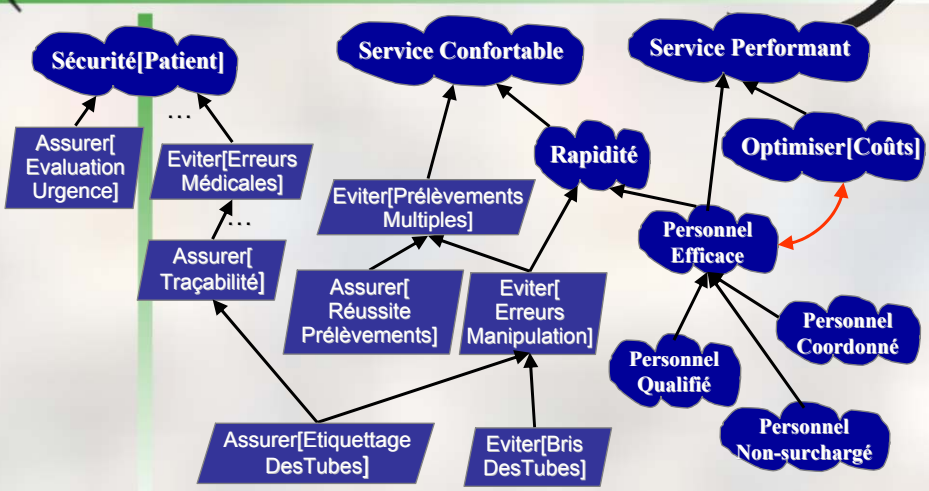
i* - contexte plus général SU et Hôpital



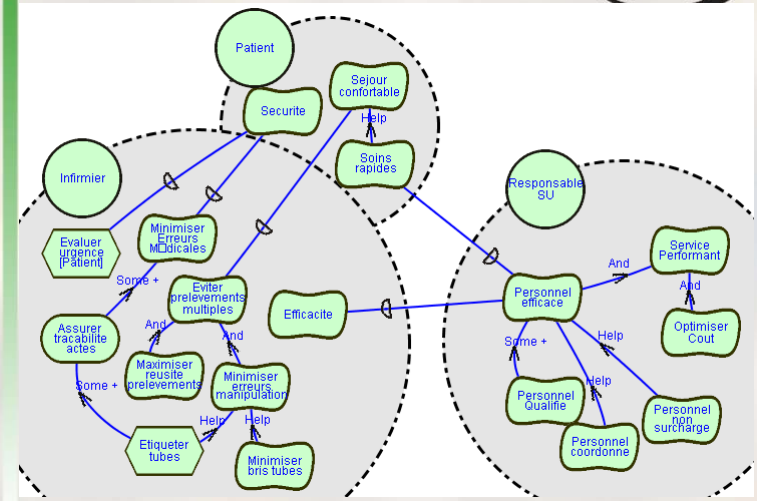
KAOS - contexte plus général SU & Hôpital



KAOS: Buts non-fonctionnels



i* : Buts non-fonctionnels



Identification de problèmes potentiels dans le système

Quelques scénarios problèmatiques

- Le médecin **oublie d'écrire au tableau blanc** qu'une PS est à faire. Il ne s'en aperçoit que plus tard, il en résulte un délai.
- L'infirmier **ne colle pas** tout de suite **des étiquettes sur les tubes** prélevés au patient. Après un moment, on ne sait plus à qui les tubes ont été prélevés. On doit alors **les jeter**. Il faut alors **refaire les prélèvements**. Dans le pire des cas, on colle des étiquettes avec un nom de **patient erroné**, ce qui peut avoir des conséquences désastreuses sur le diagnostic et le traitement qui s'en suivra.
- Le laboratoire peut **refuser une demande** si les tubes reçus ne sont pas étiquetés ou si le **nom du patient** sur l'étiquette est **illisible.** Dans le meilleur des cas, il faudra alors refaire une PS, ce qui résulte en un **délai supplémentaire**.
- Certains patients pour lesquels une PS est à faire **attendent** parfois **un temps excessif** avant que leur PS ne soit réalisée.

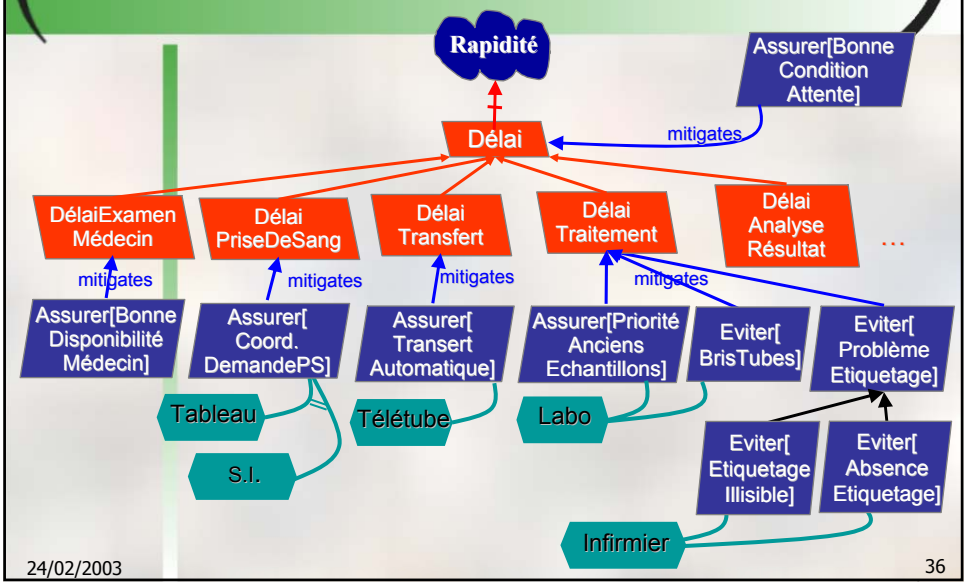
KAOS : Analyse d'obstacle

- Comment découvrir systématiquement des variantes non-souhaitables et que l'on n'a **pas** spécialement observées ?
  - Génération d'obstacles par « régression » dans les connaissance sur le domaine
  - Techniques de raffinement d'obstacles
- Comment les traiter pour en empêcher l'apparition, en diminuer l'impact, en fonction de l'importance de celui-ci ?
  - Elimination: substitution d'agent, de but, prévention
  - Tolérance: restauration du but, mitigation des conséquences
  - Ignorer

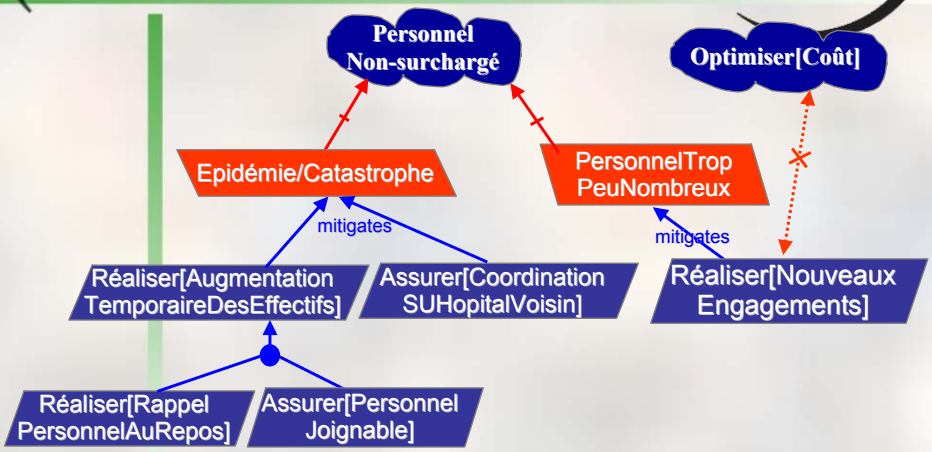
Quelques propriétés du domaine

- Les personnes oublient surtout quand elles sont surchargées.
- L'écriture humaine (et des docteurs en particulier) n'est pas toujours très lisible.
- La majorité des humains n'aiment pas attendre.
- La majorité des humains n'aiment pas les prises de sang.
- Les tubes sont fragiles et peuvent se briser lors des manipulations notamment lors de la centrifugation.
- Les urgences sont difficiles à prévoir.
- …

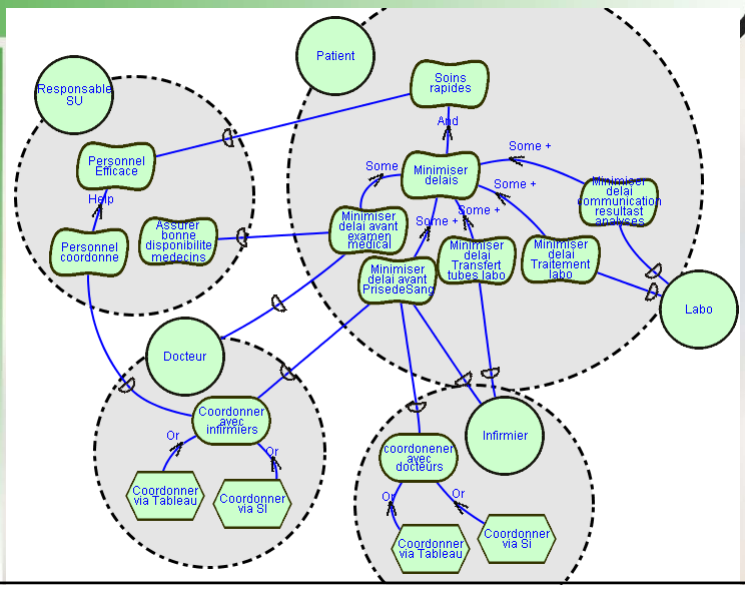
KAOS - Obstacles : délais



KAOS - Obstacles : surcharge du personnel



i* - Obstacles: délais

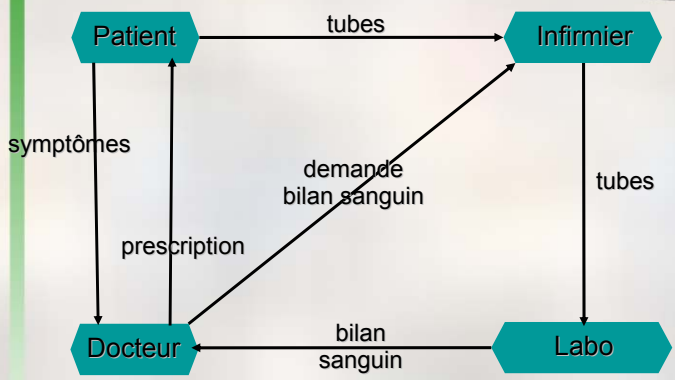


Aspects spécifiques
Outils

i*: aspects spécifiques

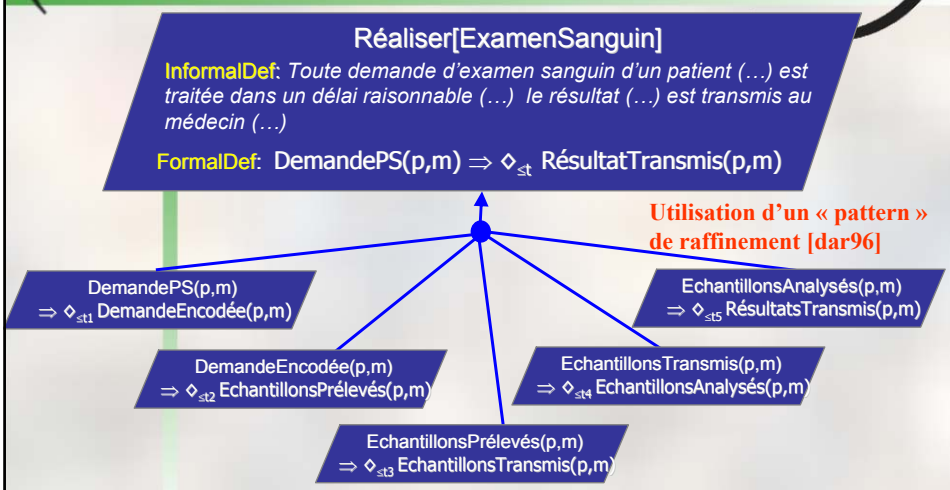
- Différentes classes d'acteurs:
  - Agent: acteur physique identifiable
  - Rôle: rôle joué par un agent
  - Position: groupe de rôles
- Procédure de labellisation [Chung99]: marquer les nœuds comme satisfaits ou pas
  - Valeurs (plus ou moins satisfait, conflit, …)
  - Propagation en fonction du type de lien
  - Interactive!
- Base de GRL (Proposé pour standardisation (ITU-T) dans le cadre de URN)

KAOS - Modèle des interfaces



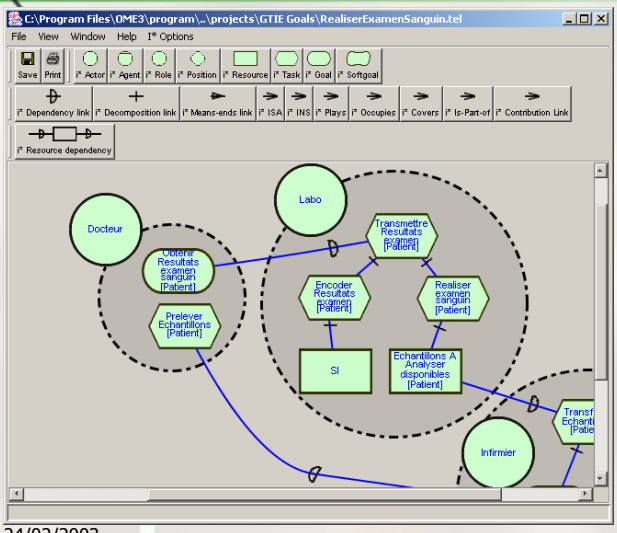
Système PS: réponse à un « manque de monitorabilité » pour que le docteur réalise le diagnostique du patient

KAOS – niveau formel



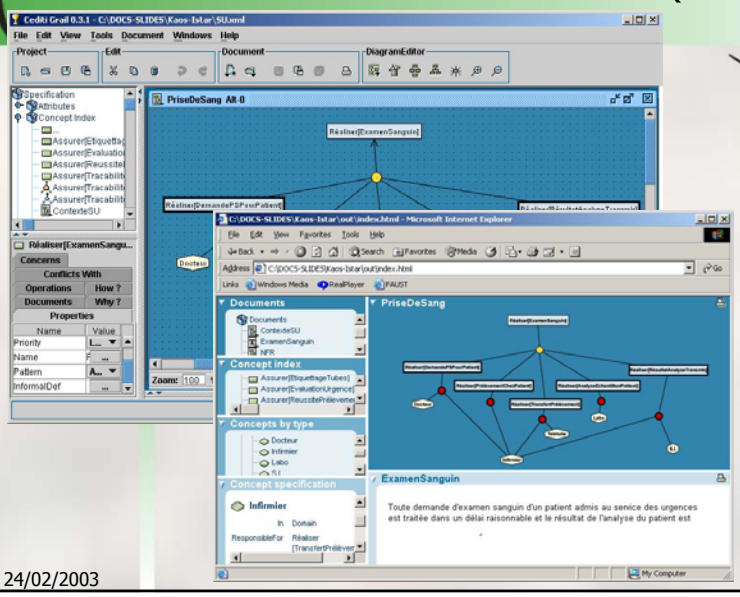
(Pour tout p: Patient, m:Médecin)

Outils i*: Organisational Modelling Environment (OME)



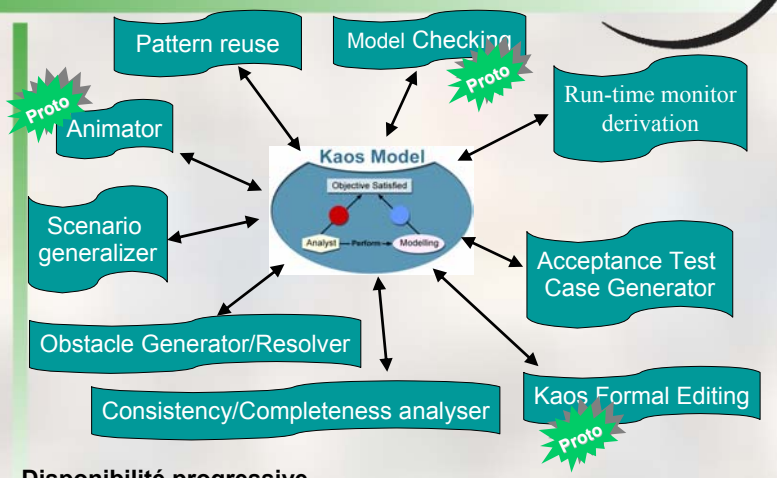
- Java
- Générique
- Labellisation
- Analyse des dépendances
- Académique

Outil KAOS : GrailLite (CEDITI-SA)



- Java
- Générique
- Génération de rapports
- **Pré-commercial** (CEDITI SA)

KAOS: atelier formel FAUST



Disponibilité progressive
Prochain Milestone en mai 2003

Synthèse

Comparaison i* / KAOS (1)

	i*	KAOS
Buts	But Soft But/Tâche	But Soft But - ontologie plus **riche** Requisite, Reqmnt, Expectation,…
Agents	Acteur **ontologie plus riche** Agent, Rôle, Position	Agent
Vocabulaire	Ressource	Objet (E-R-A)
Responsabilités	Implicite: élément à l'intérieur de l'acteur	Lien de responsabilité explicite

Comparaison i* / KAOS (2)

	i*	KAOS
Liens (% buts)	décomposition, moyens-fin, contribution dépendances Plus orienté **NFR**	ET/OU, concerns, responsibility capability, performance,… Patterns: jalon, cas → **bib.** Plus orienté **FR**
Obstacles		Obstacle, mitigation
Formalité	Semi-formel	Semi-formel → Formel
Optique	Centré sur l'acteur (vue stratégique, dépendances)	Centré sur les objectifs (cahier des charges)
Objectif	Early requirements **Organisation**	Early → Late requirements **Cahier des charges**

Approches basées sur les buts: Résumé

- Capturer le pourquoi (« rationales », motivations) → fournit les abstractions pertinentes
- Buts fonctionnels et non-fonctionnels
- Guidance méthodologique:
  - Raffinements (décompositions), alternatives
  - Opérationalisation
  - Conflits, Priorités, Compromis
  - Analyse d'obstacles
- Approches « Top-down » vs « Bottom-up »
- Informel → semi-formel → formel
- Similitudes et complémentarités de **i*** et **KAOS**

Bénéfices de l'approche buts

- Meilleure qualité du CdC
- Meilleure justification du système informatique
- Meilleure adéquation aux besoins organisationnels
- Meilleure appropriation du système par les acteurs de l'organisation

Références

- I*
  - Page de i*/OME: http://www.cs.toronto.edu/km/istar/
  - Chung and al, Non-functional requirements in software engineering, Kluwer, 1999.
- KAOS
  - Page KAOS: http://www.info.ucl.ac.be/research/projects/AVL/ReqEng.html
  - Intro: *Goal-Oriented Requirements Engineering: A Guided Tour*
  - Obstacles: *Handling Obstacles in Goal-Oriented Requirements Engineering*
  - Outil GrailLite: http://www.cediti.be

Suite de la discussion