

RESEARCH OUTPUTS / RÉSULTATS DE RECHERCHE

Analyse organisationnelle et ingénierie des exigences

Petit, Michaël; Rousseau, Anne; Schobbens, Pierre-Yves; Lobet-Maris, Claire

Published in:

ARTHUR - Manuel d'informatisation des urgences hospitalières, ISBN 2-930344-32-6

Publication date:

2003

Document Version

le PDF de l'éditeur

[Link to publication](#)

Citation for pulished version (HARVARD):

Petit, M, Rousseau, A, Schobbens, P-Y & Lobet-Maris, C 2003, Analyse organisationnelle et ingénierie des exigences: préalables à l'informatisation des services des urgences. dans J Herveg & A Rousseau (eds), *ARTHUR - Manuel d'informatisation des urgences hospitalières*, ISBN 2-930344-32-6. Presses universitaires de Louvain, Louvain-la-Neuve, pp. 13-53.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

MODULE « GARANTIE DE FONCTIONNEMENT ORGANISATIONNEL ET LOGICIEL »

CHAPITRE 1 ANALYSE ORGANISATIONNELLE ET INGENIERIE DES EXIGENCES : PREALABLES A L'INFORMATISATION DU SERVICE DES URGENCES (SU)

Dr ANNE ROUSSEAU
Prof. Dr CLAIRE LOBET-MARIS

*Cellule interfacultaire de
Technology Assesment (CITA)*

*Facultés universitaires Notre-
Dame de la Paix*

Namur

clo@info.fundp.ac.be

Dr MICHAËL PETIT
**Prof. Dr PIERRE-YVES
SCHOBENS**

*Laboratoire d'ingénierie des
exigences du logiciel (LIEL)*

*Facultés universitaires Notre-
Dame de la Paix*

Namur

pys@info.fundp.ac.be

INTRODUCTION

Dans ce premier chapitre, nous présentons une **méthodologie pluridisciplinaire et participative d'élicitation des exigences** pour la conception d'un système informatique ainsi que les leçons de son application dans le cadre de la conception d'un système d'information informatisé pour les services d'urgences hospitalières. Cette étape est un préalable incontournable de l'informatisation qui **débouche sur la rédaction du cahier des charges** du système à concevoir.

Cette méthodologie combine des éléments d'approches provenant de deux champs disciplinaires que sont l'ingénierie des exigences du logiciel (IE) et la théorie des organisations (TdO), raison pour laquelle nous parlerons d'ingénierie organisationnelle des exigences (IOE). Elle a été appliquée dans le cadre du projet ARTHUR pour les phases initiales du processus de définition des exigences d'un système informatique de support à l'activité du SUH. Cette application servira d'illustration pour différentes étapes proposées.

Dans un premier temps, nous présentons brièvement les approches qui fondent cette méthodologie (IE et TdO) pour ensuite développer les raisons pour lesquelles cette analyse des exigences organisationnelles est un préalable important à l'informatisation, entre autres des services d'urgences. Nous identifions ensuite les grandes étapes de la méthodologie proposée et, dans la foulée, les critères de qualité recommandés. Nous illustrons ensuite ces divers éléments par des exemples concrets issus de la démarche suivie dans le projet ARTHUR. Et enfin, nous précisons quel doit être le contenu général d'un cahier des charges pour un système informatique de support à l'activité d'un service d'urgences.

EN QUOI CONSISTE L'ANALYSE DES EXIGENCES ORGANISATIONNELLES ?

L'analyse des exigences organisationnelles combine donc des éléments issus de deux champs disciplinaires différents que sont l'ingénierie des exigences du logiciel et la théorie des organisations¹.

¹ Voir à ce propos [30].

Depuis plusieurs années maintenant, **l'ingénierie des exigences du logiciel** est reconnue comme une discipline clef pour assurer le succès d'un projet informatique (voir par exemple [1, 2]) et a fait l'objet de nombreux travaux. L'IE a pour objectif d'élaborer un **cahier des charges** contenant une description, la plus précise et complète possible, des exigences que doit remplir un logiciel à développer. Une exigence est définie dans le standard IEEE Std-610 [29] comme étant :

1. une condition ou capacité dont un utilisateur a besoin pour résoudre un problème ou atteindre un objectif ;
2. une condition ou capacité qui doit être satisfaite ou possédée par un système ou composant de système afin de satisfaire un contrat, un standard, une spécification, ou tout autre document imposé formellement ;
3. une représentation documentée d'une condition ou capacité telle que définie en 1 ou 2.

Le cahier des charges est élaboré à partir de besoins souvent exprimés de manière floue ou imprécise par un certain nombre de parties prenantes à un projet d'informatisation. Il doit être compris et validé par l'ensemble de ces parties prenantes.

L'IE est un « processus systématique de développement d'un ensemble d'exigences grâce à un processus coopératif et itératif d'analyse du problème, de documentation des observations résultantes dans divers formats et de vérification de l'exactitude de la compréhension obtenue des exigences » [3].

Il débute lorsque la décision est prise de recourir à un système informatique pour rencontrer tout ou partie des objectifs de l'organisation. Cette décision résulte généralement d'une analyse informelle de l'organisation et de ses objectifs au terme de laquelle le sentiment est acquis qu'un système informatique est susceptible d'améliorer le fonctionnement de l'organisation ou de permettre de saisir des opportunités nouvelles. La phase d'IE consistera donc en un processus systématique visant à préciser les opportunités ressenties d'utilisation de logiciels pour les transformer en un ou plusieurs cahiers des charges contenant des exigences précises pour un ou plusieurs logiciels à développer.

C'est précisément pour l'identification de ces opportunités que le recours à la TdO s'avère opportun : il s'agit pour l'essentiel, grâce aux outils de l'analyse organisationnelle, d'identifier les « points de rupture » du fonctionnement existant de l'organisation à informatiser en vue de cerner les potentiels d'amélioration de celui-ci pour, *in fine*, cerner ceux qui sont susceptibles d'être satisfaits par le recours à l'informatique.

La phase d'IE se termine lorsque les exigences contenues dans le cahier des charges sont définies de manière suffisamment précise et non ambiguë et qu'un accord existe, entre toutes les parties prenantes, sur ces exigences et leur adéquation vis-à-vis des objectifs de l'organisation et des motivations de ses membres [3]. Il importe donc particulièrement de s'assurer que l'ensemble des exigences imposées au logiciel à développer rencontrent les objectifs et préoccupations des acteurs de l'organisation, faute de quoi le projet informatique s'expose à des risques d'échec élevés. Le recours à l'analyse organisationnelle vise également dans ce cas de figure à apporter des éléments de réponse à cette préoccupation.

Le propre de la méthodologie proposée dans ce manuel est de combiner différentes grilles de lecture de l'organisation proposées par la TdO ainsi que des techniques d'élicitation fournies par l'IE. L'objectif de cette combinaison tient en l'acquisition d'informations générales et particulières sur le domaine d'application (en l'occurrence les services d'urgences hospitalières) et sur les exigences (besoins, attentes et contraintes des utilisateurs potentiels et de l'organisation) relatives au système à développer. Par l'application de cette méthodologie, on déduit progressivement, à partir des informations récoltées grâce aux grilles de lecture de l'analyse organisationnelle, des exigences plus formalisées et consignées dans un cahier des charges.

Comme nous allons le démontrer par la suite, la combinaison de ces deux points de vue complémentaires permet d'obtenir une vision plus complète de la problématique avec pour conséquence une **meilleure adéquation du système informatique défini aux préoccupations organisationnelles**. Le point de vue de l'analyse organisationnelle permet de mettre en perspective les exigences imposées au logiciel. De plus, la méthodologie repose sur une approche participative et active des acteurs de l'organisation, ce qui a pour avantage une probabilité accrue d'acceptation du produit logiciel final par ceux-ci (à la fois en termes d'adaptation de l'outil à l'organisation et d'appropriation de l'outil par l'organisation).

POURQUOI L'ANALYSE DES EXIGENCES ORGANISATIONNELLES EST-ELLE UN PRÉALABLE IMPORTANT À L'INFORMATISATION ?

Dans [25], une étude sur les erreurs réalisées dans l'ingénierie du logiciel montre que les erreurs commises pendant l'IE sont les plus persistantes (parmi les erreurs faites pendant la spécification, trois quarts d'entre elles ne sont corrigées qu'après l'implantation) et les plus coûteuses à corriger (les erreurs de spécification représentent 66% du coût total de correction des erreurs). Une autre étude plus récente [1] montre que, dans les projets informatiques réussis, 30% des facteurs de succès sont liés à des aspects relatifs aux exigences et à leur bonne compréhension alors que, dans les projets problématiques, les questions relatives aux exigences représentent plus de 40% des facteurs de difficulté et d'échec. Ces études montrent donc que, bien que souvent perçu comme non immédiatement rentable, l'effort porté sur la phase d'IE est effectivement récupéré en évitant des corrections coûteuses par la suite. De la même façon, l'implication des utilisateurs dès la phase de conception du système à maintes fois été avancée dans la littérature relative à l'innovation technologique et au changement organisationnel comme un facteur clef de succès [33] [34] [35].

Par ailleurs, les services d'urgences hospitalières représentent un lieu d'implantation particulier pour les systèmes informatiques. La plupart d'entre eux utilisent peu l'informatique comme support à leurs activités (tout au moins pour la partie médicale). Du fait de la nature de leur activité, les futurs utilisateurs, qui sont pour l'essentiel les médecins et les infirmiers, ne tolèrent généralement l'informatique que si elle leur permet d'exercer leur art (« l'art de guérir dans des situations d'urgence ») sans leur imposer de contraintes artificielles et expriment généralement des exigences fortes quant à la convivialité du logiciel. Il importe donc de prendre en compte la « réceptivité » des personnels soignants en ce qui concerne l'utilisation de l'outil informatique dans le cadre de leur travail quotidien. Faute de ne l'avoir suffisamment fait, quelques projets d'informatisation des urgences ont connu de cuisants échecs qui se traduisent très concrètement par la non-utilisation d'un logiciel coûteusement implanté.

EN QUOI CONSISTE UNE DÉMARCHE D'ANALYSE DES EXIGENCES ORGANISATIONNELLES ?

La méthodologie présentée ici possède deux traits caractéristiques :

- elle est **pluridisciplinaire** et combine des concepts et techniques de deux disciplines, la théorie des organisations et l'ingénierie des exigences du logiciel ;
- elle se veut **participative** et vise à impliquer fortement les acteurs concernés par l'informatisation de l'organisation.

La méthodologie préconise l'élaboration d'un cahier des charges en **quatre grandes étapes**.

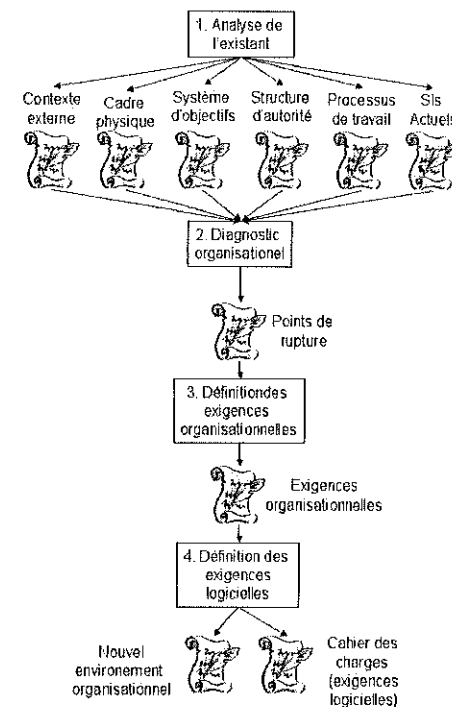


Figure 1 - Une démarche d'Ingénierie organisationnelle des exigences en quatre étapes

Étape 1 - Analyse de l'existant

La première étape consiste en une étude de l'existant. Elle vise à étudier, comprendre et décrire l'organisation telle qu'elle existe, selon le double éclairage de l'ingénierie des exigences et de la théorie des organisations. Ce double éclairage induit une analyse du domaine d'application qui se focalise sur des aspects spécifiques et produit différentes descriptions (voir figure 1).

Une première description concerne le contexte externe de l'organisation (son environnement, ses relations avec celui-ci et les changements qui y sont à l'œuvre). Ensuite, on s'attache à décrire le contexte interne de l'organisation et plus particulièrement la structure d'autorité de l'organisation (qui décide de quoi, qui gère quelle information, etc.), le cadre physique du travail (disposition et taille des locaux, localisation des systèmes d'information existants, etc.) les principaux processus de travail (qui fait quoi quand, etc.), le système de buts ou d'objectifs de l'organisation (les objectifs qui sous-tendent le système actuel et qui servent à la fois à expliquer la raison de l'existence de l'organisation et son mode de structuration et de fonctionnement), les systèmes d'informations existants (supports, types d'informations traitées, etc.).

Étape 2 - Diagnostic organisationnel

Un diagnostic organisationnel est ensuite posé sur cet existant. Il vise à évaluer dans quelle mesure l'existant rencontre de manière satisfaisante ou non les objectifs identifiés. Ce diagnostic se traduit par l'identification de **points de rupture** dans le fonctionnement de l'existant organisationnel, c'est-à-dire des éléments susceptibles d'être modifiés en vue de mieux répondre aux objectifs poursuivis. Ce diagnostic se base sur une évaluation de la réalisation des objectifs qui peut être tant qualitative que quantitative.

Étape 3 : Définition d'exigences organisationnelles

À partir de ce diagnostic sont ensuite dérivées des exigences organisationnelles qui seront traitées au cours du projet en vue d'améliorer la rencontre des objectifs de l'organisation. Ces exigences organisationnelles peuvent être définies, de façon similaire à la définition donnée à la section 2, comme des conditions ou capacités qui doit être satisfaites ou possédées par l'organisation (sa structuration, son fonctionnement, etc.). Elles sont exprimées indépendamment du recours ou non à la technologie informatique et expriment des changements envisagés de l'organisation actuelle afin d'améliorer la réalisation des objectifs insuffisamment rencontrés ou d'atteindre de nouveaux objectifs.

Étape 4 - Définition de la frontière d'informatisation

Cette dernière étape consiste en la définition d'un ensemble des possibles pour l'informatisation d'une partie des activités de l'organisation. Cette étape poursuit deux objectifs complémentaires et indissociables :

- dériver une description aussi précise que possible du rôle du logiciel (exprimé sous forme d'exigences fonctionnelles ou non fonctionnelles qui lui sont imposées) ;
- définir les hypothèses faites sur la nature et le comportement de l'environnement du logiciel (l'organisation) et les changements imposés à celui-ci par rapport à la situation courante.

Il s'agit ici de définir les rôles respectifs, quant à la rencontre des exigences organisationnelles et des contraintes d'utilisabilité du logiciel à développer, d'une part, et de l'organisation dans laquelle ce logiciel opérera, d'autre part. Tout comme Jackson [9], nous arguons que ces deux types d'exigences sont tous deux importants. Ils sont en effet nécessaires si l'on veut s'assurer que les exigences et contraintes organisationnelles définies à l'étape précédente sont bien rencontrées. En effet, ni les exigences imposées au logiciel, ni les changements organisationnels seuls ne permettront d'atteindre les objectifs. La combinaison des deux sera nécessaire.

DES REPRÉSENTATIONS À DIVERS NIVEAUX DE DÉTAIL

La démarche en quatre étapes présentée ci-dessus peut être appliquée pour obtenir des descriptions plus ou moins détaillées des divers aspects représentés sur la figure 1. Par exemple, dans le projet ARTHUR, elle a été appliquée deux fois, lors de **deux phases principales**.

- Une première phase d'étude globale du contexte d'implantation du système informatique a permis de cerner ce que nous appelons les pistes d'informatisation. Les descriptions produites étaient relativement abstraites et ne tenaient pas compte de tous les détails des SUH étudiés.
- Une seconde phase a étudié de manière plus détaillée chaque opportunité d'informatisation identifiée au terme de la première phase. Les descriptions produites lors de cette phase sont beaucoup plus précises et incluent de nombreuses informations détaillées, par exemple sur les processus de travail étudiés.

La première phase d'étude globale poursuit le double objectif :

- d'étudier et de comprendre l'organisation dans laquelle le système informatique va s'insérer (dans notre cas, le service d'urgences hospitalier, sa composition, son fonctionnement, ses objectifs, le système de valeur de ses membres, etc.) et
- d'identifier globalement les opportunités d'amélioration de l'organisation au moyen d'un ou plusieurs systèmes informatiques.

Cette première étape est à nos yeux importante à deux titres :

- tout d'abord afin d'assurer que les informaticiens participant au projet (développeurs et ingénieurs des exigences) possèdent une bonne connaissance du domaine d'application (l'organisation cible) ;
- ensuite, afin de permettre aux diverses parties prenantes de bien percevoir les besoins et opportunités d'amélioration qu'offre un système informatique. Cette démarche est d'autant plus importante que ces besoins ou opportunités sont au départ difficilement identifiables pour les informaticiens (ingénieurs des

exigences) participants au projet, mais aussi pour les acteurs du domaine eux-mêmes.

Étant donné les objectifs de cette phase, il n'est pas indispensable d'obtenir des représentations complètement détaillées des aspects étudiés.

L'objectif spécifique de la deuxième phase (études détaillées) consiste à approfondir l'analyse de chacune des opportunités identifiées à la première étape, afin de dériver, in fine, un cahier des charges précis pour chacune d'entre elles. Cet objectif spécifique justifiait, quant à lui l'utilisation de descriptions précises et détaillées.

Dans tous les cas de figure, il y a lieu d'adapter le niveau de détail au contexte du projet et en particulier à l'objectif poursuivi en appliquant la démarche, aux contraintes de temps et de budget imposées au projet, etc.

UN PROCESSUS PARTICIPATIF ET CRÉATIF

Nombreux sont les recherches et travaux qui ont mis en exergue la nécessaire implication des utilisateurs dans la conception même des SI comme facteur clef de succès de ces projets informatiques (par exemple, [1,13]). La méthodologie décrite ici satisfait également ces impératifs de participation des professionnels de l'urgence.

Le processus d'IOE est idéalement un processus d'apprentissage coopératif et participatif dans lequel les membres de l'organisation sont impliqués. Nous privilégions donc une approche participative et coopérative dans laquelle les acteurs de l'organisation concernés par le nouveau système informatique à concevoir collaborent à l'élaboration des exigences. Cela présente entre autres l'avantage de limiter à terme la survenance de conflits et problèmes ultérieurs tels que, par exemple, la non-utilisation, voire le sabotage du système par les utilisateurs finaux.

QUI DOIT PARTICIPER À L'IOE ?

Il importe que tous les acteurs de l'organisation concernés par le nouveau système informatique soient impliqués dans le processus d'IOE. Il convient donc avant toute chose d'identifier les acteurs

concernés et de mettre en place des mécanismes qui leur permettent de participer à la définition des exigences.

D'une façon générale, les participants potentiels à l'activité d'IOE incluent au minimum (voir figure 2) :

- les gestionnaires responsables ou décideurs de l'organisation (ou à tout le moins des unités organisationnelles concernées par le projet de système informatique) : ils sont la source d'expression des objectifs de l'organisation et, plus spécifiquement, des objectifs à assigner au projet informatique. Par exemple, dans le cas d'informatisation d'un service d'urgences hospitalier, il peut s'agir des responsables du service (médecin-chef de service et infirmier-chef) et/ou de l'hôpital (directeur général, directeur médical ou nursing, etc.) ;
- les utilisateurs potentiels : ils sont une des sources principales des exigences car ce sont eux qui connaissent le mieux le système actuel, ses qualités et défauts, et les tâches qui sont susceptibles d'être informatisées. Il est opportun de distinguer plusieurs classes d'utilisateurs dans le cas où des groupes d'utilisateurs potentiels ont des besoins spécifiques. En fonction du logiciel à développer, on pourra par exemple avoir des médecins, des infirmiers, du personnel d'accueil, du personnel de secrétariat, du personnel logistique, etc. mais aussi des personnes ne relevant pas du service des urgences mais échangeant des informations importantes avec celui-ci tels que les différents services techniques d'examen et d'analyse (radio, labo, etc.).
- divers spécialistes, et notamment :
 - les spécialistes en théorie des organisations : ils étudient l'organisation et ses modes de fonctionnement en détail et posent un diagnostic sur ceux-ci ;
 - les ingénieurs des exigences : ils sont responsables de la collecte, de la synthèse, de la documentation et de la gestion des exigences et de leur traduction en cahier(s) des charges ;
 - les ingénieurs du logiciel : ils seront responsables du développement du logiciel sur base du cahier des charges. Il importe donc qu'ils participent à la phase d'IOE afin de valider la faisabilité du logiciel défini dans le cahier des charges et de bien comprendre les exigences imposées au logiciel ;

- des juristes, spécialisés dans le domaine du droit médical : connaissant les contraintes légales relatives au domaine médical, ils sont en mesure de fournir des exigences à respecter (par exemple relatives aux droits du patient, au secret médical, aux contraintes imposées par la sécurité sociale, etc.) ;
- des spécialistes en diverses technologies dont l'utilisation est pressentie pour la construction du logiciel (par exemple, la cryptographie, les cartes à puces, la reconnaissance vocale, les réseaux sans fil, les PDAs, etc.).

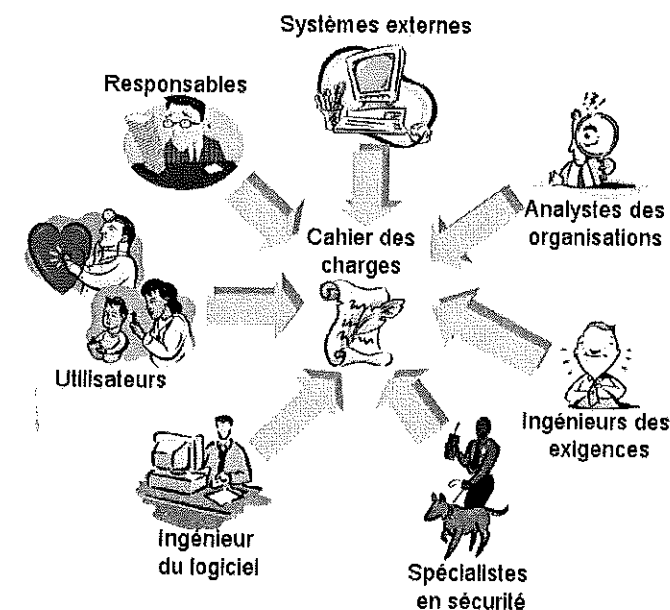


Figure 2 - Les participants à l'IOE

En règle générale, il y a lieu d'impliquer dans le processus d'IOE tous les utilisateurs qui ont potentiellement une interaction avec le logiciel ou sur le travail desquels le logiciel aura un impact. En particulier, si le logiciel doit avoir des interactions avec d'autres logiciels existants, le point de vue de ces logiciels devra être pris en compte. Cela implique donc la participation au processus d'IOE d'acteurs responsables de ces systèmes. Par exemple, si l'on définit un cahier des charges dans le cadre du développement d'un système logiciel

pour informatiser l'envoi d'un formulaire de demandes d'analyses biologiques au laboratoire, il sera nécessaire d'impliquer des représentants du laboratoire dans la démarche d'ingénierie des exigences, à plus forte raison si le logiciel doit être interfacé avec les systèmes logiciels propres au laboratoire. De même, comme la plupart des développements de systèmes médicaux hospitaliers se font dans le cadre d'une informatique centralisée existante, l'implication des services informatiques responsables de cette infrastructure sera donc nécessaire.

Dans certains cas, il peut être intéressant d'impliquer également des représentants des patients.

Le nombre de parties prenantes peut être relativement important ce qui peut être source de problèmes. Les souhaits, besoins et attentes exprimés par différentes personnes peuvent en effet être contradictoires et il devient alors difficile pour l'ingénieur des exigences de faire des choix parmi les nombreuses exigences contradictoires. Il est opportun dans ce cas d'élaborer une structure de représentation et de participation de ces parties prenantes avec des règles de décision claires (par exemple, par vote) pour le traitement des exigences contradictoires et la gestion des priorités entre exigences. Nous préconisons le recours aux conférences de consensus pour ce faire.

COMMENT ÉLABORER DES MODÈLES DE QUALITÉ ?

La démarche de l'IOE présentée ci-dessus implique, à chacune de ses étapes, la création de plusieurs représentations (modèles) de différents aspects importants durant l'IOE, schématisés à la figure 1, à savoir une représentation de divers aspects du système existant (son contexte externe, son système d'objectif, etc.), la description de points de rupture, les exigences organisationnelles, les exigences logicielles (cahier des charges) et le nouvel environnement organisationnel.

Afin d'élaborer chacune de ces descriptions, un microprocessus itératif d'interaction avec les parties prenantes est nécessaire à chaque étape. Les activités de ce microprocessus de l'IOE sont représentées sur la figure 3. Elles sont ensuite décrites plus en détail dans les sections suivantes.

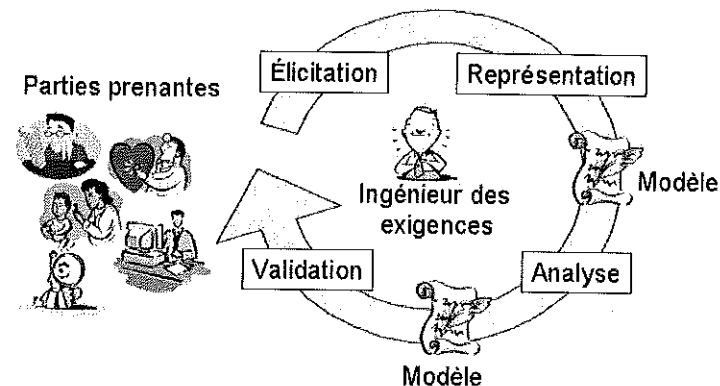


Figure 3 - Le cycle d'élaboration de modèles

L'élicitation

L'activité d'élicitation a pour objectif d'acquérir auprès des parties prenantes l'information nécessaire lors de la phase d'IOE. Elle peut avoir recours à diverses techniques, dont :

- des entretiens : les entretiens² avec les parties prenantes peuvent être plus ou moins structurés, en fonction du type d'information désiré. Il peut s'agir d'entretiens individuels ou collectifs qui peuvent être ouverts (discussion libre, *brainstorming*) ou plus ou moins dirigés (sur base de questionnaires ou check-lists préétablis) ;
- des observations : l'analyste observe les procédures existantes de travail et de traitement des informations en interférant le moins possible avec elles³. L'utilisation d'enregistrements audio ou vidéo constitue également une approche intéressante étant donné la rémanence qu'ils procurent ;

² L. ALBARELLO, F. DIGNEFFE, J.-P. HERNIAUX, C. MAROY, D. RUQUOY et P. DE SAINT-GEORGE, *Pratiques et méthodes de recherche en sciences sociales*, Paris, Armand Colin, 1995.

³ « ...l'observation consiste à se trouver présent, mêlé à une situation sociale pour l'enregistrer et l'interpréter en s'efforçant de ne pas la modifier » in H. PERETZ, *Les méthodes en sociologie : l'observation*, Paris, La Découverte, 1998, p 5.

- l'analyse documentaire : l'analyste peut également étudier divers types de documents en vue d'acquérir de la connaissance utile à l'IOE. Par exemple : des documents décrivant les procédures de travail actuellement utilisées, des textes légaux, des standards, etc. ;
- l'analyse de logiciels existants : les logiciels actuellement utilisés dans l'organisation sont une source évidente d'information à la fois pour une analyse de l'existant et pour la définition des exigences. Des logiciels existant sur le marché et fournissant des fonctionnalités similaires peuvent également servir de source à la définition des exigences par analogie.

La représentation

L'activité de représentation consiste à consigner l'information acquise lors de l'élucidation dans des modèles explicites. Les modèles sont exprimés dans un ou plusieurs langages. Un **langage** fournit un ensemble de symboles et une grammaire définissant des règles valides d'utilisation et de composition de ces symboles pour former des modèles.

On distingue plusieurs **catégories de langages** en fonction de leur degré de précision (formalité) :

- les langages naturels (par exemple, le français) ;
- les langages graphiques ou semi-formels : ils fournissent un ensemble limité de concepts associés à des symboles graphiques. Par exemple, les schémas Entités/Association ou les diagrammes de cas d'utilisation d'UML (use case diagrams) ;
- les langages formels : outre qu'il fournissent un ensemble de concepts limités, la sémantique de ces langages (la signification des concepts qu'ils fournissent) est définie de manière rigoureuse sur base de formalismes mathématiques. Par exemple, la logique des prédicats du premier ordre, les réseaux de Petri ou les diagrammes état-transition (state charts).

Chacune de ces catégories de langages possède ses avantages et ses inconvénients. Il y a donc lieu de faire un **choix du langage utilisé** pour définir les modèles en fonction des caractéristiques souhaitées dans une situation particulière. On peut par exemple comparer les langages en fonction des **critères** qui suivent .

- L'ambiguïté : c'est la possibilité d'interprétation (de compréhension) de manière différente d'un modèle par des lecteurs différents (ou par un seul lecteur à des moments différents). Le langage naturel est le moins précis et donc le plus ambigu. Les langages semi-formels sont plus précis car ils ne permettent l'expression que d'un nombre plus restreint de modèles et que leur sémantique est généralement définie de manière rigoureuse. Les langages formels, quant à eux, ne laissent aucune place à des divergences d'interprétation.
- Le caractère naturel : c'est la facilité qu'a le lecteur d'exprimer ou de comprendre un modèle. Les modèles semi-formels graphiques sont généralement les plus naturels et sont parfois préférés au langage naturel (« un petit dessin vaut mieux qu'un long discours »). Les langages formels sont en général moins accessibles pour des non-spécialistes.
- L'expressivité : c'est le degré de couverture par le langage de l'ensemble des choses que l'auteur du modèle veut exprimer. Le langage naturel est le plus expressif. Il ne possède pratiquement pas de limite à ce qui peut être exprimé. L'expressivité des langages semi-formels et formels dépend des concepts qu'ils renferment mais est toujours plus limitée que celle du langage naturel.
- La possibilité d'analyse et de traitement automatique : plus un langage est formel, plus il permettra un traitement automatisé important des modèles. Par exemple, un modèle graphique pourra être analysé pour vérifier s'il respecte les règles de grammaire qui lui sont imposées. Les langages formels sont ceux qui permettent le plus d'analyses et de traitements. Il est par exemple possible avec certains langages de réaliser une simulation en exécutant le modèle.

Certaines approches tentent de combiner les avantages de plusieurs catégories de langages. Par exemple, des approches basées sur des structures prédéfinies de textes en langage naturel tentent de conserver l'avantage de l'expressivité tout en introduisant plus de structure, pour rendre le traitement automatique plus aisé. C'est le cas, par exemple, des canevas de cahiers des charges [11,18,19] ou encore des structures de description de cas d'utilisation (scénarios) [20].

Certains langages formels sont associés à des symboles graphiques afin de les rendre plus naturels et compréhensibles (par exemple, les

réseaux de Petri ou les diagrammes état/transition). En pratique, vu l'effort de formalisation important requis par l'utilisation des langages formels, on en limitera souvent l'utilisation à certains types de systèmes (dits critiques) qui nécessitent un degré de sûreté élevé et dans lesquels l'ambiguïté doit être évitée. Par exemple, un logiciel de contrôle d'un appareillage médical dont le dysfonctionnement serait potentiellement dangereux pour le patient (comme un pousse-seringue qui administre automatiquement un médicament à un patient) se prêtera bien à l'utilisation d'un langage formel.

L'analyse de la représentation/modèle

Une fois représentée dans un modèle, l'information élicitée par l'ingénieur des exigences peut être analysée afin de s'assurer d'un certain nombre de qualités intrinsèques du modèle. Les analyses peuvent être plus ou moins automatisées en fonction du degré de formalité du langage. Un modèle en langage naturel ne permettra que peu d'analyses et sera généralement traité manuellement.

Les **qualités d'un modèle** que l'analyse vise à vérifier sont notamment [3] :

- la correction syntaxique : le modèle doit être syntaxiquement bien formé (les « phrases » qui composent le modèle doivent respecter les règles propres au langage) ;
- la cohérence : le modèle ne contient pas de contradictions ;
- la complétude : tous les éléments utilisés dans le modèle sont définis de manière complète ;
- la non-ambiguïté : le modèle est précis et ne peut être interprété de manière incorrecte ;
- la non-redondance : le modèle évite de répéter plusieurs fois les mêmes éléments ou de fournir explicitement des éléments qui peuvent être dérivés à partir d'autres.

La validation de la représentation/modèle

La validation est une étape importante de l'IOE. Elle consiste à s'assurer que le modèle réalisé représente de manière juste et complète l'information exprimée par les parties prenantes. Elle vise à s'assurer de **l'adéquation entre le modèle et la réalité**.

Pour ce faire, **plusieurs techniques** peuvent être employées, comme par exemple :

- la présentation des modèles aux parties prenantes : les modèles sont exposés aux parties prenantes. En fonction du type de langage utilisé pour écrire le modèle et de la connaissance des parties prenantes, un paraphrasage (ré-expression du modèle dans un langage plus naturel) peut être nécessaire.
- le prototypage : une maquette du système informatique est élaborée sur base du modèle défini jusqu'ici et est présentée à l'utilisateur pour obtenir un retour d'information sur son adéquation par rapport à ses besoins ;
- l'animation ou la simulation : certains modèles, écrits dans des langages formels, peuvent être interprétés ou exécutés de telle manière qu'il est possible de montrer aux parties prenantes des exemples de comportements du système. Il peut alors être demandé aux parties prenantes de confirmer ou d'infirmer la possibilité de tels comportements dans le système réel.

ÉLABORATION DE CAHIERS DES CHARGES : EXEMPLES CONCRETS

Dans la suite, nous allons présenter quelques exemples de mises en œuvre des processus et techniques décrits ci-dessus. Ils illustrent comment ces techniques peuvent être utilisées pour élaborer et valider des descriptions utiles pour les différentes étapes de la méthodologie présentée à la figure 1. Les exemples concernent tous l'informatisation d'un service d'urgences hospitalier.

Exemple 1 - Élicitation et représentation de processus actuels de travail sur base de scénarios

Comme il est indiqué à la figure 1, l'élaboration d'un cahier des charges commence par une analyse de l'existant. Une partie de cette analyse concerne l'étude et la représentation des procédures actuelles de travail. Plusieurs techniques peuvent être utilisées pour analyser cet aspect de l'existant. Dans ce premier exemple, nous montrons comment une approche basée sur les scénarios a été utilisée dans le cadre de l'étude du processus général de prise en charge du patient dans un SUH.

Tout d'abord, les acteurs susceptibles de participer à cette étude ont été identifiés. Comme le centre d'intérêt était les procédures de travail actuelles, les acteurs étaient principalement les ingénieurs des exigences et toutes les personnes prenant une part active au processus de prise en charge, c'est-à-dire le personnel médical, le personnel infirmier, le personnel administratif, le responsable du SUH et le personnel logistique. Bien que le patient soit au centre de ce processus, il n'a pas semblé opportun de l'inclure dans le processus car celui-ci n'a souvent pas une connaissance complète et suffisante du processus suivi lors de sa prise en charge.

La représentation des procédures actuelles avait pour objectif une compréhension générale des activités liées à la prise en charge du patient afin de pouvoir par la suite l'évaluer au vu des objectifs identifiés comme ceux du SUH (étape de diagnostic organisationnel).

Élicitation

Pour l'élicitation de l'information relative aux procédures actuelles de travail, outre l'analyse de documents tels que des textes légaux, des descriptions des procédures disponibles pour certaines tâches spécifiques, la description de logiciels actuellement utilisés et divers formulaires utilisés dans le SUH, c'est l'observation qui a principalement été utilisée. Les ingénieurs des exigences ont observé *in situ* le processus de prise en charge du patient sur un certain nombre de cas concrets.

Une approche basée sur les scénarios et les cas d'utilisation [20] a été privilégiée. Dans ces approches, un scénario représente un exemple de comportement possible d'un acteur ou ensemble d'acteurs et est constitué d'une suite ordonnée d'actions réalisées par les acteurs ou d'interactions entre ceux-ci. Un scénario permet donc de représenter de manière très intuitive un comportement observé ou possible d'un groupe d'acteurs.

Cette approche a été mise en œuvre au moyen de grilles d'observation dans lesquelles les ingénieurs des exigences ont noté systématiquement les activités et échanges d'informations des professionnels de l'urgence. Trois types de grilles ont été utilisées à cette fin :

- une *grille orientée patient*, relative au cheminement du patient dans l'hôpital lors de sa visite au SUH ;
- une *grille orientée intervenant*, relative aux activités des différentes catégories de personnels (personnel infirmier,

administratif ou médical) participant à la prise en charge du patient pendant une période d'activité du SUH ;

- une *grille orientée échanges de données*, dans laquelle étaient notées les informations générées, échangées, stockées et/ou modifiées (par exemple, un dossier patient, le tableau de suivi des patients, etc.) pendant une période d'activité dans le SUH.

Représentation

À partir des grilles décrites ci-dessus, des scénarios typiques de prise en charge du patient ont été identifiés. De ces scénarios a été induite une description du processus général de prise en charge du patient qui représente l'ensemble de ces scénarios typiques. Le processus obtenu a été représenté au moyen des diagrammes d'activité UML [22,23,24].

Cette description graphique montre l'enchaînement des différentes activités relatives à la prise en charge du patient dans le service d'urgences. Elle montre également des alternatives envisageables dans cet enchaînement ainsi que des activités facultatives.

La figure 4 représente le modèle le plus abstrait de ce processus. L'objectif de ce modèle n'est pas d'être complètement exhaustif mais de fournir un cadre simple par rapport auquel peuvent être situées les différentes activités du service. Ainsi celles-ci sont regroupées en trois grande catégories : les activités qui concernent l'accueil, celles qui relèvent du diagnostic et enfin celles qui visent à orienter le patient c'est-à-dire faire un choix pour une poursuite adéquate des soins du patient. Une série de descriptions en langage naturel font également partie de la description des processus représentés.

Bien que ce ne fut pas notre cas, on pourrait également élaborer un modèle descriptif des informations manipulées dans le cadre de ces processus pour compléter le modèle du processus. Celui-ci peut être réalisé par exemple avec des diagrammes de classes UML (tel que celui présenté à la figure 11 dans la section 5.4).

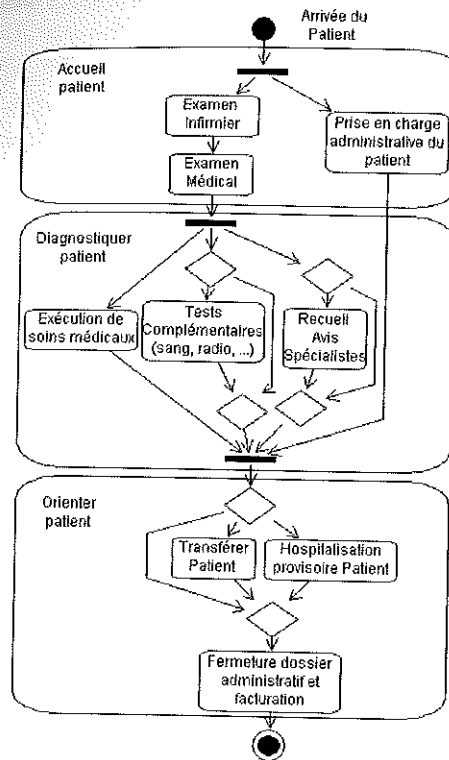


Figure 4 - Exemple de représentation des processus de travail existants :
modèle général du processus de prise en charge du patient
dans un service d'urgences

Validation

Le modèle élaboré a été validé sur base de nouvelles observations et d'interviews. À partir de l'information recueillie jusqu'alors, les ingénieurs des exigences ont posé des questions aux acteurs afin d'obtenir des informations supplémentaires sur ces procédures et les informations manipulées. Les modèles ont été modifiés et précisés en conséquence.

Exemple 2 - Élicitation et représentation des objectifs organisationnels

Le second exemple que nous utiliserons concerne l'élicitation des objectifs organisationnels des SUH. Les objectifs organisationnels représentent les objectifs et motivations des différents acteurs de

l'organisation. Ils sous-tendent et régulent les comportements de ces acteurs, aussi bien individuels que collectifs. La prise en compte de ces objectifs est importante à plus d'un titre. D'une part, elle est essentielle pour la réalisation du diagnostic organisationnel (selon le degré de réalisation de ces objectifs) ; d'autre part, elle permet d'évaluer et de justifier l'opportunité du SI à concevoir (cohérence des objectifs organisationnels et informatiques).

Les participants à l'identification de ces objectifs sont globalement les mêmes que ceux identifiés à la section 6.1. En outre, le point de vue du législateur, définissant légalement certains objectifs et contraintes relatifs aux SUH, a été pris en compte par l'implication de juristes.

Élicitation

Hormis l'étude des textes légaux définissant la mission des SUH, l'élicitation a principalement eu lieu sur base d'interviews. Au cours de celles-ci, les objectifs collectifs (du SUH dans son ensemble) et individuels (des différentes catégories de personnel) ont été identifiés en questionnant les acteurs sur plusieurs aspects. Par exemple, la question du pourquoi de certaines procédures de travail ou de choix observés dans le fonctionnement du SUH a été soulevée avec les intervenants du SUH. Ceux-ci ont été interrogés sur leur perception de leur rôle et mission parmi ceux du SUH ainsi que sur les missions pressenties du SUH dans son ensemble. De même, les problèmes de la situation actuelle, tels que ressentis par ces intervenants, ont permis d'identifier des objectifs actuellement insuffisamment satisfaits. Les interviews semi-structurées ont été préférées à des interviews trop dirigées afin de laisser émerger les motivations sans biaiser les interlocuteurs.

Représentation

Les objectifs identifiés ont tout d'abord été consignés dans un document en langage naturel. Une distinction a été faite entre les objectifs de mission (relatifs aux activités et aux patients pris en charge) et les objectifs de système (relatif à l'équipe du SUH) [14]. La figure 5 montre un extrait de cette description.

La mission première de ces services d'urgences nous a été présentée par nos interlocuteurs, d'une façon générale, comme étant celle qui consiste à accueillir, diagnostiquer et orienter le patient, quel qu'il soit. Il s'agit là de ce que nous appellerons le but de mission de l'organisation, défini par un cadre légal, centré sur l'Homme dans un souci de soins pour tous et ce, de façon équitable.

Une des contraintes fortes du système des urgences est d'être confronté à une demande hétérogène en croissance depuis quelques années. Aucune prise en charge d'une personne se présentant au SU ne peut être refusée mais le patient peut être « ré-orienté » ; il doit donc dans tous les cas de figure faire l'objet d'une attention toute particulière au vu du contexte médico-légal belge. [...]

Les buts de système, quant à eux, sont par définition quelque peu spécifiques à chaque service : le premier mettant davantage l'accent sur une certaine conception de l'efficacité du service (performances et optimisation des procédés et façons de faire), le deuxième sur une volonté de contrôle de l'environnement (la demande de soins), le troisième enfin s'inscrivant davantage dans un système visant la pérennité de l'organisation hospitalière.

Les objectifs de système se définissent également en termes de qualité des soins apportés au patient (ce qui suppose un temps d'attente réduit mais aussi une minimisation des risques d'erreur médicale et une attention toute particulière pour les relations établies avec le patient et ses accompagnateurs ainsi que pour l'aspect « hôtelier » de son séjour) et de légitimité du service au sein et en dehors de la structure hospitalière. Enfin, l'esprit d'équipe et la solidarité fonctionnelle qui prévalent dans ces services nous ont également été cités par nos interlocuteurs comme un objectif du système en tant que tel.

Figure 5 - Description en langage naturel des objectifs d'un SUH

Une partie des objectifs a ensuite fait l'objet d'une semi-formalisation dans le langage i^* [7,8]. Ce langage permet de représenter graphiquement les buts des différents acteurs et leurs relations. Ce langage est particulièrement approprié pour la représentation des motivations qui sous-tendent l'existence, le fonctionnement et la structuration de systèmes organisationnels et techniques. Un modèle i^* est composé d'acteurs dont les motivations sont représentées par un graphe composé de nœuds et de liens.

Le modèle de la figure 6 montre la formalisation d'une partie des buts d'un SUH. Dans ce cas, pour la représentation des buts partagés par l'ensemble des membres du SUH, un seul acteur représentant l'ensemble du SUH est considéré. Il est également possible de représenter plusieurs acteurs ou types d'acteurs (médecins, infirmiers, etc.) auxquels seront alors associés des objectifs spécifiques. Le graphique montre qu'un des objectifs principaux du SUH est de sauver les vies de patients dits « critiques », bien qu'un autre objectif stipule la nécessité de prendre en charge tous les patients, quelle que soit la gravité de leur cas. Apparaît également le processus de prise en charge des patients en trois étapes mis en œuvre pour atteindre ces objectifs. D'autres objectifs partagés par l'ensemble des intervenants du SUH sont également représentés, comme l'importance de l'esprit d'équipe ou l'équité de traitement des patients.

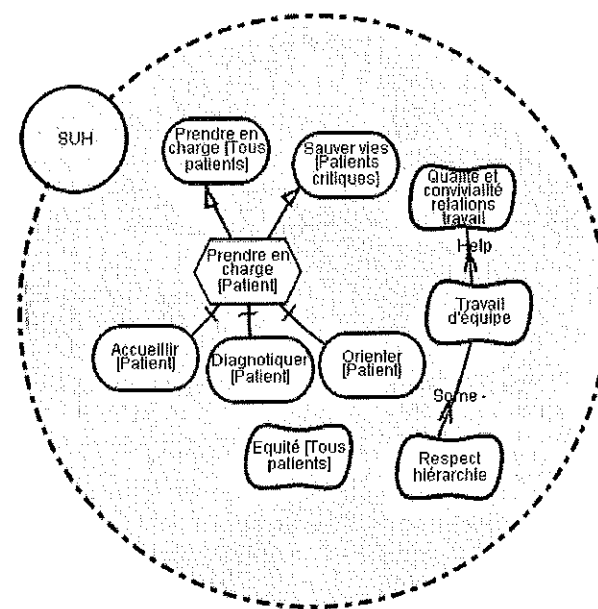


Figure 6 - Extrait du modèle de buts du SUH.

Validation

Le modèle des objectifs a été validé avec l'ensemble des intervenants par une présentation interactive à ceux-ci. La présentation a eu lieu principalement sur base d'une description textuelle des objectifs et a été faite en même temps que la présentation d'un diagnostic organisationnel (voir section 6.3 ci-dessous).

Exemple 3 - Élaboration et validation d'un diagnostic organisationnel

Sur base de diverses descriptions de l'existant, un certain nombre de points de rupture dans les modes de fonctionnement actuels des services d'urgences hospitalières peuvent être identifiés. L'identification de ces points de rupture constitue un « diagnostic organisationnel ». Les points de rupture sont autant d'éléments susceptibles d'être améliorés par le biais de l'introduction d'un système d'information informatisé dans ces services.

Bien que l'identification des objectifs décrite à la section 6.2 et le diagnostic organisationnel sont ici présentés comme se déroulant de façon chronologique, dans les faits, ils sont souvent réalisés en parallèle. Le diagnostic permet généralement d'identifier ou de préciser des objectifs non encore explicités. De même, l'identification et le raffinement des objectifs permettent également un affinement du diagnostic organisationnel.

Élicitation

Les points de rupture ont été identifiés de deux façons. Premièrement, ils proviennent des besoins et attentes exprimés par les utilisateurs potentiels du système à concevoir. Ceux-ci ont été interviewés à propos des problèmes actuels (objectifs non rencontrés de manière satisfaisante ou dysfonctionnements ressentis). Ensuite, des points de rupture ont été identifiés par les ingénieurs des exigences en opérant des analyses à partir des données collectées et au vu des objectifs précédemment identifiés.

Représentation

Dans notre cas particulier, les points de rupture identifiés étaient de cinq ordres et concernaient la gestion du flux patient, l'organisation

du travail, les systèmes d'information existants, les locaux et les relations externes du SUH.

Ils ont tout d'abord été décrits en langage naturel. La figure 7 montre un exemple de description de diagnostic relatif à la gestion du flux des patients dans le SUH.

Le travail réalisé au sein d'un SU se caractérise par une grande discontinuité : la taille des services, la disposition de locaux, le nombre de cas à traiter en parallèle et la nécessité de « faire vite et bien » sont des facteurs explicatifs de cette discontinuité. Il n'est pas rare qu'une infirmière qui procède à une prise de sang auprès d'un patient doive répondre au téléphone, administrer un médicament à un autre patient et donner à boire à un troisième avant de pouvoir envoyer les échantillons prélevés au laboratoire... Les interruptions sont donc fréquentes, ce qui implique pour le personnel de procéder en permanence à l'identification des actes à poser (savoir ce qui a été fait et ce qui est à faire) et à des arbitrages entre ceux-ci (problème de priorités). Le souci de traiter tous les patients de la même façon, d'une part et de respecter la hiérarchie médicale, d'autre part rendent ces arbitrages d'autant plus complexes. Les arrivées imprévisibles des patients et les besoins variables en termes de ressources limitées que cela implique obligent constamment le personnel à réévaluer les priorités. Le problème majeur qui se pose dès lors à chacun est de savoir comment établir les priorités entre ses interventions compte tenu du fait que ces priorités sont à tout moment susceptibles d'être remises en question. Dans le même ordre d'idées, nous avons pu constater que, vu la complexité et la diversité des tâches à réaliser, les intervenants se posent régulièrement la question de savoir ce qui a été fait et ce qui reste à faire. Il ne s'agit plus ici de savoir dans quel ordre il faut réaliser ces différentes tâches mais bien davantage de savoir lister ces tâches et identifier leur état.

De la même façon, il faudrait à tout instant pouvoir « tracer le patient » c'est-à-dire le localiser, connaître son état particulier et savoir qui le soigne. Plusieurs intervenants nous ont présenté cette difficulté comme problématique et ayant une influence néfaste sur l'efficacité du service (temps perdu) et la qualité des soins (notamment, le temps d'attente du patient). Le tableau de suivi des patients utilisé actuellement dans le SUH présente un certain nombre d'insatisfactions au regard de ces différents objectifs (...).

Figure 7 - Une partie de diagnostic organisationnel relative à la gestion du flux patient

La figure 8 montre une formalisation de ce diagnostic au moyen du langage i*. Les activités de gestion du flux des patients sont représentées par une tâche (*Gérer Flux Patients*) qui requiert notamment le tableau de suivi des patients. Le degré avec lequel certains objectifs sont rencontrés a été évalué au vu des raisonnements tenus ci-dessus. Ces évaluations sont représentées par des signes à droite des objectifs correspondants. Le signe X pointé à droite de *Facilité [identification actes...]* indique que cet objectif échoue partiellement : la facilité avec laquelle les actes à réaliser et leur priorité peuvent être identifiés est insatisfaisante. En conséquence, la qualité du processus de gestion du flux patient est jugée insuffisante, et ceci a par exemple pour conséquence que l'objectif de minimiser le temps d'attente des patients n'est que partiellement satisfait (ce qui est représenté par un V pointé). Ce dernier objectif pouvait avoir été identifié comme un des objectifs fondamentaux du SUH (objectif de système) lors de l'identification des objectifs tel que présenté à la section 6.2.

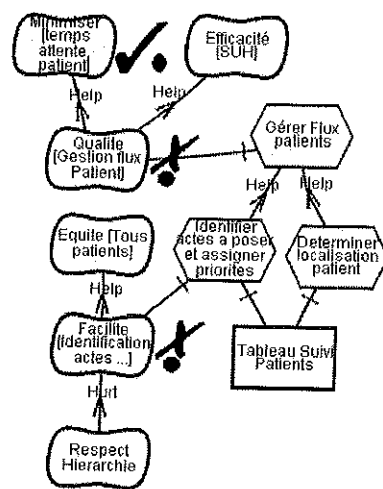


Figure 8 - Représentation i* d'une partie du diagnostic organisationnel

Validation

Le diagnostic organisationnel réalisé a été progressivement affiné par des interviews complémentaires avec les différents intervenants des SUH. Une validation du diagnostic final a été réalisée sur base d'une présentation faite à l'ensemble des membres des SUH. Cette présentation a eu lieu sur base de descriptions en langage naturel et

non sur base des modèles i* eux-mêmes et ce en raison de leur difficile accessibilité pour des non-initiés. Dans d'autres situations, l'utilisation de ces graphiques pourrait être envisagée.

Exemple 4 - Élaboration d'un cahier des charges pour le logiciel

Le dernier exemple que nous considérerons concerne l'élaboration d'un cahier des charges précis pour une partie d'un système logiciel de support à l'activité du SUH. Il illustre la seconde étape (étude détaillée) de notre méthodologie telle que présentée au point 4. La fonctionnalité envisagée concerne l'informatisation des demandes d'analyses biologiques faites en urgence au laboratoire. Pour élaborer ce cahier des charges, l'ensemble des étapes de la figure 1 ont été réalisées. Ainsi, le processus actuel de demandes d'analyses et les différents aspects associés (objectifs, contexte externe, cadre physique, SI actuels, etc.) ont été étudiés en détail, puis un diagnostic a été réalisé afin de déterminer les exigences organisationnelles à satisfaire. Les exigences logicielles ont ensuite été identifiées. Dans cette section, nous montrons comment le cahier des charges contenant les exigences logicielles a été décrit et validé (étape de définition de la frontière d'informatisation de la figure 1).

Les participants à cette dernière activité étaient les mêmes que ceux identifiés à la section 6.1. Les ingénieurs du logiciel ont en outre été impliqués et ont fortement interagi avec les ingénieurs des exigences, afin d'évaluer et d'assurer de la faisabilité d'une implémentation logicielle des exigences.

Élicitation

Les exigences du logiciel ont principalement été dérivées des exigences organisationnelles. De plus, ces exigences ont été complétées à partir d'informations acquises lors d'interviews avec les intervenants du SUH.

Représentation

Les exigences du logiciel ont été consignées dans un cahier des charges textuel structuré selon une partie de la norme PSS05, utilisée anciennement pour l'ingénierie des systèmes spatiaux, qui concerne la description des exigences logicielles [11]. Cette norme préconise une découpe du cahier des charges en sections

prédéfinies. La figure 9 montre la structure générale d'un cahier des charges conforme à cette norme.

Service Information :
a - Abstract
b - Table of Contents
c - Document Status Sheet
d - Document Change Records made since last issue
1 INTRODUCTION
1.1 Purpose
1.2 Scope
1.3 Definitions, acronyms and abbreviations
1.4 References
1.5 Overview
2 GENERAL DESCRIPTION
2.1 Relation to current projects
2.2 Relation to predecessor and successor projects
2.3 Function and purpose
2.4 Environmental considerations
2.5 Relation to other systems
2.6 General constraints
2.7 Model description
3 SPECIFIC REQUIREMENTS
(The subsections may be regrouped around high-level functions)
3.1 Functional requirements
3.2 Performance requirements
3.3 Interface requirements
3.4 Operational requirements
3.5 Resource requirements
3.6 Verification requirements
3.7 Acceptance testing requirements
3.8 Documentation requirements
3.9 Security requirements
3.10 Portability requirements
3.11 Quality requirements
3.12 Reliability requirements
3.13 Maintainability requirements
3.14 Safety requirements
4 REQUIREMENTS TRACEABILITY MATRIX
5 Annexes

Figure 9 - Structure d'un cahier des charges logiciel conforme à la norme PSS05

Outre l'utilisation du langage naturel, des langages semi-formels (plus précis) peuvent être utilisés. Dans notre cas, le langage UML a été utilisé pour la représentation de divers aspects du cahier des charges et les modèles UML (graphiques) ont été insérés dans des sections spécifiques du cahier des charges. Par exemple, des cas d'utilisation [20,23] ont été utilisés pour représenter les interactions possibles entre des utilisateurs et le système informatique.

Chaque cas d'utilisation représente une séquence normale des interactions qui prennent place lors de l'utilisation du système par l'utilisateur, ainsi que des séquences alternatives (manières différentes d'arriver au même objectif ou situations exceptionnelles pouvant se présenter).

Dans une approche basée sur les scénarios [26], l'ensemble de ces cas d'utilisation représente les fonctionnalités que le logiciel doit fournir aux utilisateurs. Un diagramme de cas d'utilisation représente cet ensemble de cas graphiquement. Il est représenté à la figure 10, où l'on voit l'interaction des différentes classes d'utilisateurs avec le système informatique. Chacun des cas d'utilisation fait l'objet d'une description plus précise. Celle-ci peut être exprimée en langage naturel comme montré à la figure 11.

Des diagrammes de séquence UML (permettant de représenter graphiquement des scénarios d'utilisation du système informatique) pourraient aussi être utilisés. La séquence normale (cas le plus fréquent ou dans lequel tout se passe bien) est décrite comme une suite d'interactions qui peuvent elles-mêmes être des interactions complexes (comme la première étape qui représente les interactions nécessaires à l'identification de l'utilisateur).

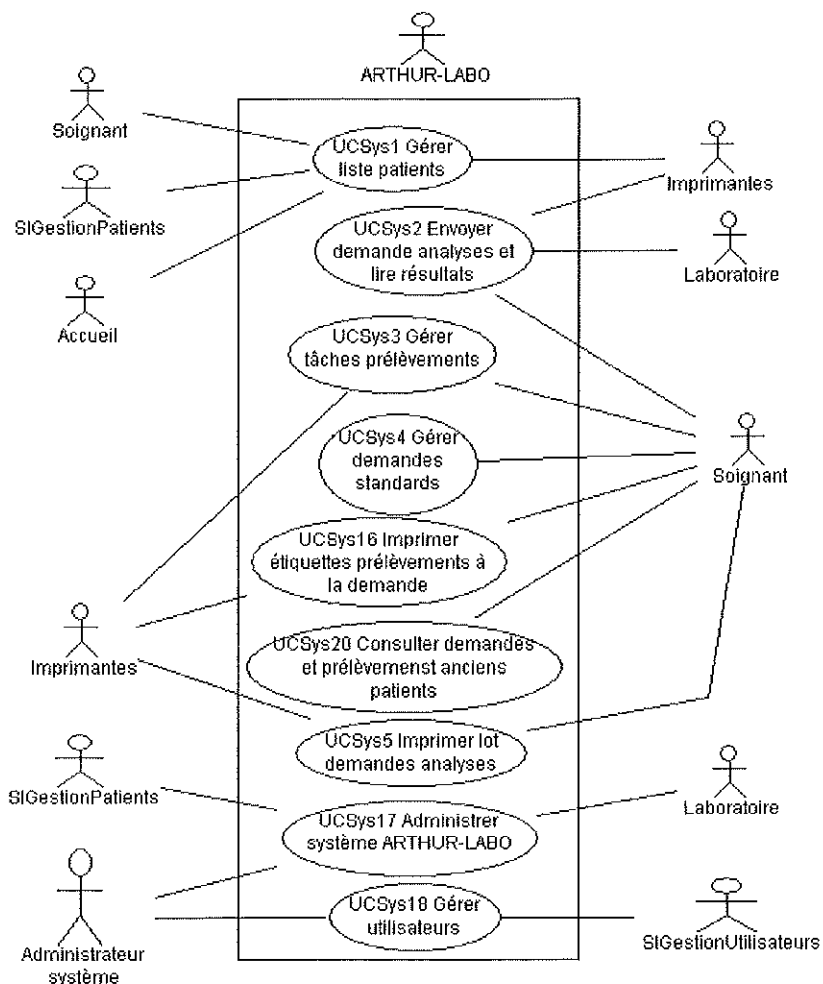


Figure 10 - Exemple de diagramme de cas d'utilisation

UCSys2 Envoyer demande analyses et lire résultats	
Objectif de base	Décrire toutes les activités relatives au remplissage d'une demande d'analyses, son envoi au laboratoire, la réalisation des analyses et à la lecture des résultats
Préconditions	ARTHUR-LABO fonctionne
Postconditions	ARTHUR-LABO fonctionne
Cas normal	1. include UCSys10 Identifier 2. include UCSys6 Créer nouvelle demande analyses 3. include UCSys8 Signer et envoyer demande analyses 4. include UCSys14 Finir session 5. Le laboratoire réalise les analyses 6. Le laboratoire prévient ARTHUR-LABO que les résultats sont disponibles 7. ARTHUR-LABO affiche que les résultats sont disponibles. 8. include UCSys10 Identifier 9. Le soignant sélectionne la demande et indique à ARTHUR-LABO qu'il veut visualiser les résultats. 10. ARTHUR-LABO déclenche l'ouverture de SI-LABO pour permettre la lecture des résultats des analyses 11. include UCSys14 Finir session
Variantes non modélisées par des Use Case	Annulations : - θ si l'étape 1 se termine par un échec, le UC s'arrête. - θ si l'étape 2 s'est terminée par une annulation, l'étape 3 n'a pas lieu et le UC s'arrête après l'étape 4 - θ si l'étape 3 s'est terminée par une annulation, le UC s'arrête après l'étape 4 - θ si annulation d'une demande, annulation des tâches de prélèvement
Variantes modélisées par des Use Case	UCSys2.1 Avec enregistrement intermédiaire et modification UCSys2.2 Avec deux soignants UCSys2.3 Avec annulation demande

Figure 11 - Exemple de spécification en langage naturel d'un cas d'utilisation

Un autre exemple d'élément important du cahier des charges est la description des informations stockées et manipulées par le logiciel. Ces informations peuvent être représentées en UML par un diagramme de classes. Un exemple de tel diagramme est présenté à la figure 12. On y voit les différents concepts (appelés classes) à propos desquels de l'information est manipulée (appelés classes) ainsi que les informations utiles (appelées attributs) à propos de chacun de ces concepts et les relations existant entre ces concepts (appelées relations). Ce genre de modèles servira généralement de base pour la définition de la structure du contenu de la base de données du logiciel.

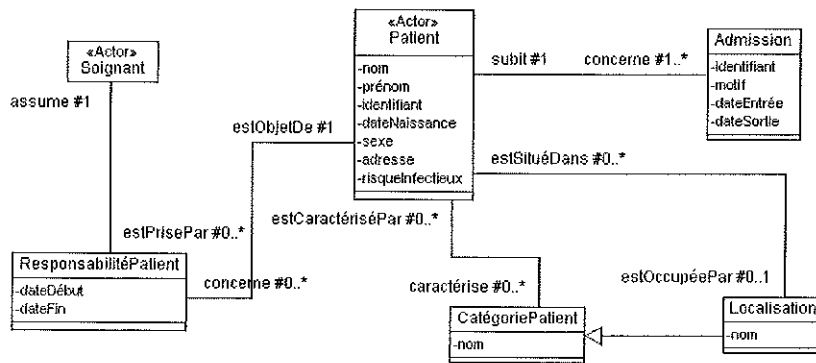


Figure 12 - Exemple de diagramme de classes représentant une partie des informations manipulées par le logiciel (informations relatives au patient et informations liées)

Validation

Le cahier des charges complet élaboré tel que défini ci-dessus a été validé au moyen de plusieurs techniques. Tout d'abord, au cours de son élaboration, des interviews ont permis de valider les directions prises dans la définition des fonctionnalités. Ensuite, une approche par prototypage, dans laquelle des implémentations préliminaires ont été présentées aux utilisateurs, a été utilisée. Les remarques des utilisateurs ont donné lieu à des modifications simultanées du prototype et du cahier des charges. Finalement, le cahier des charges complet a été relu par des représentants des différentes

catégories des utilisateurs et a fait l'objet d'une validation et acceptation finale.

QUEL DOIT ÊTRE LE CONTENU GÉNÉRAL D'UN CAHIER DES CHARGES POUR UN SYSTÈME INFORMATIQUE DE SUPPORT À L'ACTIVITÉ D'UN SERVICE D'URGENCES ?

Un cahier des charges typique pour un système d'information dédié aux SUH doit contenir un certain nombre d'informations. Nous en donnons ici une liste non exhaustive et une description sommaire.

- Les exigences fonctionnelles

Elles concernent les fonctionnalités du logiciel à développer, ce qu'il doit faire ou permettre de faire, ses services rendus aux utilisateurs. Par exemple, permettre la gestion des informations relatives au « passage » d'un patient au service d'urgences, imprimer des étiquettes, envoyer une demande d'analyses au laboratoire,...

- Les contraintes légales et de certification

Les lois et règlements à respecter lors de la mise en place de logiciels de soins de santé. Ceux-ci incluent des recommandations comme celles élaborées par la Commission Normes en matière de Télématicque au service du secteur des Soins de Santé [21], des contraintes légales (cf. chapitre 2), des normes de certification applicables aux logiciels médicaux (par exemple [27,28]), etc.

- Les mécanismes de sécurité à mettre en œuvre

Le cahier des charges doit décrire différents niveaux de sécurité nécessaires et les mécanismes de contrôle d'accès à mettre en place (identification par mot de passe, authentification par carte à puces, biométrie,...) pour permettre et restreindre l'accès aux fonctionnalités et données aux différentes catégories de personnes adéquates (cf. chapitre 3).

- Les contraintes socio-économiques ayant une influence sur les fonctionnalités du logiciel.

Par exemple, les contraintes imposées par des organismes publics tels que l'INAMI pour autoriser le remboursement de prestations.

- Les contraintes d'ergonomie du logiciel

La nature du travail médical peut imposer des contraintes fortes sur les interfaces homme-machine du logiciel (possibilité de travail en position debout, écrans tactiles, nombre d'interactions réduit, utilisation de PDAs, etc.).

- Les contraintes de traçabilité des actes et modifications

Ces exigences portent sur la nature des actes et utilisations du logiciel (actes médicaux, infirmiers, gestion du dossier patient, etc.) qui doivent pouvoir être retracés (qui, quand, où, pourquoi, etc.).

- Les contraintes de disponibilité et de robustesse (24h/24, permanence et disponibilité de l'information, procédures de récupération en cas d'incident, de résistance à la charge de travail, etc.).
- Les contraintes d'efficacité (temps de réponse, etc.)
- Les exigences en termes d'ouverture du logiciel et d'interfaçage avec d'autres logiciels
- Les éventuels standards à respecter (standards de structuration de dossier patient, de communication, etc.).

CONCLUSIONS

Ce premier chapitre du manuel d'informatisation des urgences décrit l'activité d'élaboration de cahier des charges pour un système d'information informatisé relatif aux SUH. Cette activité importante doit être menée avec soin si l'on veut garantir le succès du projet d'informatisation. Nous avons décrit une démarche basée sur des bonnes pratiques qui peut être suivie, ainsi que des techniques d'acquisition, de représentation, de traitement et de validation des exigences utiles pour cette démarche. La démarche proposée présente l'originalité de combiner des techniques usuelles d'ingénierie des exigences avec celles de l'analyse organisationnelle.

L'intérêt majeur de cette façon de faire réside dans le caractère plus global de l'approche, qui ne se limite pas aux aspects fonctionnels du

(des) logiciel(s) à concevoir. Ce dépassement est entre autres permis par l'implication forte des utilisateurs dans l'identification des exigences organisationnelles mais aussi par la prise en compte de leurs motivations et habitudes de travail. Cette façon de faire permet donc de prendre en considération les besoins et attentes des utilisateurs tout en les aidant à les formuler explicitement. Par ailleurs, cette méthodologie permet également de ne pas se « contenter » d'informatiser l'existant mais bien de l'améliorer, en accord avec les principaux intéressés, et de le compléter par un « regard externe » en termes de diagnostic organisationnel. De la sorte, il est possible de poursuivre conjointement un objectif dit « technologique » de performance de l'outil informatique et un objectif plus « social » d'appropriation par les professionnels de l'urgence concernés.

Bien que celle démarche soit le plus souvent réalisée par des experts/consultants externes à l'organisation, il est tout à fait possible qu'une partie de la démarche soit entreprise par les acteurs de l'organisation elle-même. L'objectif de la présentation faite ici est de permettre aux décideurs d'un projet d'informatisation de maîtriser les enjeux de cette démarche. Il nous semble en effet que, même si la démarche est dirigée par des experts/consultants extérieurs à l'organisation, les décideurs doivent rester attentifs à un certain nombre de préoccupations identifiées dans ce manuel, dont notamment :

- l'implication de l'ensemble des parties prenantes au processus (décideurs, utilisateurs, services externes concernés, patient, etc.) ;
- le sérieux de la démarche et des outils utilisés pour l'élaboration du cahier des charges (techniques et effort d'élucidation, de représentation, d'analyse et de validation des descriptions produites) ;
- la qualité du cahier des charges final qui doit faire l'objet d'un accord entre le « client » (le commanditaire du logiciel à développer) et le « fournisseur » (le développeur et fournisseur du logiciel) et est souvent le résultat d'un compromis entre la qualité désirée et l'effort consacré à son élaboration ;
- la complétude du cahier des charges en prenant en compte non seulement des aspects fonctionnels mais également des nombreuses contraintes notamment légales et d'utilisabilité (efficacité, ergonomie, etc.) ;

- l'utilité et l'importance de la représentation explicite (dans des modèles) de différents aspects de l'organisation et de leur utilisation dans une démarche de qualité, qui vise non seulement l'introduction d'un logiciel mais également l'amélioration continue de l'organisation. Par conséquent, il importe également que l'organisation elle-même reste maître de ces modèles c'est-à-dire qu'elle y ait un accès aisé et puisse les utiliser dans le futur, indépendamment de l'intervention ponctuelle des experts/consultants et de l'identité des experts/consultant.

RÉFÉRENCES

- [1] STANDISH GROUP INT'L, *The CHAOS report*,
http://www.pm2go.com/sample_research/chaos_1994_1.asp
- [2] R. H. THAYER and M. DORFMAN (eds), *System and Software Requirements Engineering*, IEEE Computer Society Press, 1990.
- [3] P. LOUCOPOULOS and V. KARACOSTAS, *System Requirements Engineering*, Mc Graw-Hill Book Company, 1995.
- [4] A. DE VOS, A. ROUSSEAU et CL. LOBET-MARIS, « From co-ordination to loyalty, Developing a constructivist perspective on organisational and technological changes », *18th EGOS colloquium*, Barcelone, 6-8 juillet 2002.
- [5] M. J. HATCH, *Théorie des organisations*, De Boeck Université, Bruxelles, 2000.
- [6] W. J. ORLIKOWSKI et S. R. BARLEY, « Technology and Institutions : What Can Research on Information Technology and Research on Organizations Learn from Each Other ? », in *MIS Quarterly*, vol. 25, n°2, june 2001, pp.145-165.
- [7] E. S. K. YU, *Modelling Strategic Relationships for Process Reengineering*, PhD thesis, Computer Science Department, University of Toronto (Canada), 1994.
- [8] i* web site, <http://www.cs.toronto.edu/km/istar/>, january 2003.
- [9] M. JACKSON, *Software Requirements & Specifications, A lexicon of practice, principles and prejudices*, Addison-Wesley, 1995.
- [10] K. FORSBERG and H. MOOZ, « System Engineering Overview », In [THA90a], pp. 44-72.
- [11] European Space Agency, *ESA software engineering standards - Issue 2, ESA PSS-05-0 Issue 2*, February 1991.
- [12] IFIP-IFAC TF on Architectures for Enterprise Integration, GERAM : Generalised enterprise reference architecture and methodology. Version 1.6.3, Available at <http://www.cit.gu.edu.au/~bernus/taskforce/geram/versions/>, March 1999.
- [13] K. L. MCGRAW and K. HARBISON, *User-Centered Requirements*, Laurence Elerbaum Associates, 1997.

- [14] H. MINTZBERG, *Structure et dynamique des organisations*, Ed. d'Organisation, Paris, 1982.
- [15] S. GHERNAOUTI-HÉLIE, *Stratégie et ingénierie de la sécurité des réseaux*, Paris, InterEditions, 1998,
- [16] L. DUSSEY, H. DUCROT et F.-A. ALLAERT, *L'information médicale. L'ordinateur et la loi*, 2^e éd., Éditions Médicales Internationales, 1999.
- [17] J.-F. CHANLAT, « L'analyse des organisations, un regard sur la production de langue française contemporaine (1950-1990) », in *Cahier de recherche HEC Montréal*, n°91-09, Mar. 1991.
- [18] IEEE Recommended Practice for Software Requirements Specification, IEEE Std 830-1998, IEEE, New York, 1998, also appears in [2]
- [19] J. ROBERTSON and S. ROBERTSON, *Volere Requirements Specification Template*, Edition 8, Atlantic Systems Guild Limited, <http://www.systemsguild.com/GuildSite/Robs/Template.html>
- [20] A. COCKBURN, *Writing Effective Use Cases. The Agile Software Development Series*, Addison Wesley, 2001.
- [21] Service public fédéral santé publique, Sécurité de la chaîne alimentaire et environnement, Commission normes en matière de télématique, au service du secteur des soins de santé, site web : <http://www.health.fgov.be/telematics/>
- [22] M. FOWLER and K. SCOTT, *UML Distilled. A brief guide to the standard object modeling language*, 2^d edition, Object Technology series, Addison-Wesley, 1999.
- [23] G. BOOCH, J. RUMBAUGH and I. JACOBSON, *The unified modeling language user guide*, Object Technology series, Addison-Wesley, 1999.
- [24] Object Management Group, UML website, <http://www.omg.org/technology/uml>
- [25] B. BOEHM, *Software Engineering Economics*. Prentice-Hall Inc., 1981.
- [26] I. JACOBSON, G. BOOCH and J. RUMBAUGH, *The Unified Software Development Process*, Addison-Wesley Longman, 1999.
- [27] Législation communautaire, Directive 93/42/CEE du Conseil du 14 juin 1993, relative aux dispositifs médicaux, Journal officiel n° L 169 du 12/07/1993, Document 393L0042, pp. 0001 - 0043

- [28] IEC SC 62A (COMMON ASPECTS OF ELECTRICAL EQUIPMENT USED IN MEDICAL PRACTICE), IEC 60601-1 - Ed. 2.0 : Medical electrical equipment - Part 1 : General requirements for safety, December 1988.
- [29] IEEE Standard Glossary of Software Engineering Terminology (IEEE Std610.12-1990), IEEE Computer Soc., Dec. 10, 1990.
- [30] A. ROUSSEAU and M. PETIT, « Enriching a Requirements Elicitation Methodology with an Organisational Sciences perspective : An Application in the Health Care domain », soumis à *l'International Conference on Requirements Engineering*, RE'03, September 2003, Monterey, USA.
- [31] S. GHERNAOUTI-HÉLIE, *Stratégie et ingénierie de la sécurité des réseaux*, Paris, InterEditions, 1998.
- [32] L. DUSSEY, H. DUCROT et F.-A. ALLAERT, *L'information médicale. L'ordinateur et la loi*, 2^e éd., Éditions Médicales Internationales, 1999.
- [33] M. AKRICH, M. CALLON et B. LATOUR, « À quoi tient le succès des innovations ? Premier épisode : l'art de l'intéressement », in *Annales des Mines, Gérer & Comprendre*, juin 1988, pp. 4-17.
- [34] H. BARKI, S. RIVARD et J. TALBOT, « Risque, mode de gestion et succès d'un projet d'informatisation », in *Technologies de l'information et société*, vol. 5, n°2, 1992, pp. 121-146.
- [35] E. MUMFORD, « Participative Systems Design : Practice and Theory », in *Journal of Occupational Behavior*, vol. 4, 1983, pp 47-57.
- [36] J. FULK et G. DESANTIS, « Electronic Communication and Changing Organizational Form », in *Organization Science*, vol. 6, n°4, 1995, pp 337-349.