

**APPROCHE À BASE DE VÉRIFICATION FORMELLE
DE MODÈLE POUR UNE UTILISATION SÉCURITAIRE
DE LA CUISINIÈRE D'UN HABITAT INTELLIGENT**

par

Thibault de Champs

Mémoire présenté au Département d'informatique
en vue de l'obtention du grade de maître ès sciences (M.Sc.)

FACULTÉ DES SCIENCES

UNIVERSITÉ DE SHERBROOKE

Sherbrooke, Québec, Canada, 8 avril 2012



Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 978-0-494-91674-2

Our file Notre référence

ISBN: 978-0-494-91674-2

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Le 11 avril 2012

*le jury a accepté le mémoire de Monsieur Thibault De Champs De Saint Leger
dans sa version finale.*

Membres du jury

Professeur Bessam Abdulrazak
Directeur de recherche
Département d'informatique

Professeure Hélène Pigot
Codirectrice de recherche
Département d'informatique

Professeur Marc Frappier
Évaluateur interne
Département d'informatique

Professeur Luc Lavoie
Président rapporteur
Département d'informatique

Sommaire

Le vieillissement provoque des modifications des facultés physiques et cérébrales de la personne. Ce déclin est parfois source pour elle de difficultés à maintenir son rythme et ses conditions de vie habituelles et conduit alors à une perte d'autonomie qui nécessite une prise en charge. Actuellement, cette aide est offerte par l'intermédiaire d'aidants. Cependant des technologies telles que celles développées au laboratoire DOMUS de l'Université de Sherbrooke tendent à améliorer et faciliter le maintien à domicile.

Pour s'assurer que les personnes âgées soient en sécurité au domicile, le projet INOVUS s'intéresse aux risques liés à l'utilisation de la cuisinière. Dans le cadre de ce projet, les travaux de M.Sc. présentés dans ce mémoire se concentrent sur la perspective logicielle de la détection et de la prévention des risques physiques pour la personne, lors de la réalisation de tâches utilisant la cuisinière.

Dans un premier temps, une revue des risques à domicile recensés dans la littérature a permis de définir la couverture nécessaire à une telle solution. Certaines situations dangereuses ont ensuite été sélectionnées pour définir un modèle de solution satisfaisant.

Le développement d'une solution de sécurité pour la personne entraîne des contraintes de fiabilité de très haut niveau pour la technologie produite. Pour répondre à ce besoin, la proposition de ces travaux de M.Sc. est l'utilisation de spécifications formelles. Ces outils permettent d'obtenir un plus haut degré de fiabilité de logiciels. En se basant sur ces outils, un modèle de solution a été élaboré pour le projet INOVUS, et ce à l'aide du vérificateur de modèle ALLOY.

Enfin, une implémentation en Java de ce prototype a été réalisée afin d'évaluer les résultats de détection des situations dangereuses. Ce prototype permet alors à la

SOMMAIRE

fois de valider l'approche de développement choisie, ainsi que d'établir une preuve de concept d'une telle solution de sécurité.

Mots-clés: habitats intelligents ; informatique pervasive ; sécurité ; risques à domicile ; spécifications formelles ; vérification de modèles ; inovus.

Remerciements

Je tiens tout d'abord à remercier mes directeurs de recherche, Bessam Abdulrazak et H el ene Pigot, pour m'avoir accompagn e tout au long de ce projet. Ces deux ann ees de travaux ont  et e pour moi l'occasion de me forger une exp erience dans le domaine de la recherche scientifique, et cela n'aurait pas  et e possible sans leur supervision.

Je tiens aussi  a remercier toute l' equipe du laboratoire Domus pour m'avoir fourni un environnement de travail  a la fois s erieux, motivant et convivial. En particulier, un grand merci  a Patrice C. Roy pour m'avoir offert aussi souvent son aide, notamment en ce qui concerne la r edaction d'articles scientifiques, ou encore L^AT_EX. Merci de plus  a Jean-Paul Viboud pour avoir  et e pr esent au moindre souci technique, machine  a caf e incluse.

De mani ere plus g en erale, mes remerciements s'adressent  a toutes les personnes que j'ai eu l'occasion de rencontrer  a l'universit e et qui sont intervenues d'une mani ere ou d'une autre dans le d eroulement de ma ma ıtrise.

Merci  a ma famille et mes amis, qu eb eois et fran ais, qui m'auront apport e le soutien moral n ecessaire tout au long ces deux ann ees, particuli erement dans les p eriodes difficiles. J'ai ainsi pu maintenir la motivation et l' energie n ecessaire pour compl eter ce programme.

Merci aux membres de mon jury, Luc Lavoie, Marc Frappier, ainsi que mes directeurs de recherche. Les excellents conseils re us aussi bien lors de l' evaluation de ce

REMERCIEMENTS

mémoire que dans les étapes qui l'ont précédées auront apporté énormément à ce projet, ainsi qu'à moi-même.

Pour terminer, merci à toutes les personnes que je n'ai pas mentionnées mais qui sont intervenues de plus ou moins proche lors de mon séjour à Sherbrooke. Cette expérience restera pour moi riche d'enseignements, professionnellement et personnellement parlant.

Et puis enfin, un remerciement tout spécial aux adeptes du tarot du laboratoire Domus pour avoir occupé la majorité des pauses méridiennes, je sais maintenant grâce à Pierre-Yves que même sans jeu, tout est possible... ou pas.

Abréviations

INOVUS Intelligent Oven @ Université de Sherbrooke

DTA Démence type Alzheimer

TCC Traumatisme Cranio-Cérébral

AVQ Activité de la Vie Quotidienne

PwSN People with Special Needs

ADL Activity of Daily Living

OSGi Open Service Gateway initiative

ABRÉVIATIONS

Table des matières

Sommaire	iii
Remerciements	v
Abréviations	vii
Table des matières	ix
Liste des figures	xiii
Liste des tableaux	xv
Liste des programmes	xvii
Introduction	1
1 Revue de la littérature	13
1.1 Généralités sur les risques à domicile	14
1.1.1 Entrée en matière	14
1.1.2 Définitions	14
1.1.3 Risques identifiés dans la littérature	15
1.2 Gestion des risques à domicile	20
1.2.1 Prévention et sensibilisation	21
1.2.2 Évaluations de l'habitat	21
1.2.3 Technologies d'assistance	23
1.2.4 Technologies spécifiques à l'utilisation de la cuisinière	26
	ix

TABLE DES MATIÈRES

1.3	Positionnement du projet INOVUS	28
1.3.1	Synthèse sur les risques à domicile	28
1.3.2	Apports du projet INOVUS	32
1.4	Usage de la spécification formelle	33
1.4.1	Utilisations dans des domaines connexes à la couche logicielle du projet INOVUS	34
1.4.2	Utilisation pour la couche logicielle du projet INOVUS	34
2	Approche de conception logicielle	37
2.1	Introduction	41
2.2	INOVUS Intelligent Oven Project	42
2.2.1	Context	42
2.2.2	Scenario	42
2.2.3	Specifications	43
2.2.4	Approach	44
2.3	Related Works	45
2.3.1	Consistency Issue in Software Development Process	46
2.3.2	Formal Specifications	46
2.4	INOVUS Development Process	48
2.4.1	First Increment Specifications and Design	48
2.4.2	Implementation and Analysis Using ALLOY	49
2.5	Results and Discussion	50
2.5.1	Relevancy of the Proposed Process	50
2.5.2	Evaluation of ALLOY in Pervasive Computing	51
2.6	Conclusion	52
2.7	Compléments sur l'utilisation du vérificateur de modèle	53
2.7.1	Motivation	53
2.7.2	Construction du modèle formel	54
2.7.3	Phase de vérification	55
2.7.4	Code ALLOY produit	56
3	Validation d'approche à l'aide d'un prototype	57
3.1	Introduction	61

TABLE DES MATIÈRES

3.2	Home Safety	63
3.2.1	Home Safety Evaluations	63
3.2.2	Current Technological Coverage for Home Safety	64
3.2.3	Smart Homes	66
3.3	INOVUS Concept	66
3.3.1	INOVUS purposes	66
3.3.2	INOVUS Architecture	67
3.3.3	Model Checking in INOVUS development process	69
3.3.4	From Formal Specification to Code Implementation	71
3.4	Software Implementation	72
3.4.1	System Architecture	72
3.4.2	Code Architecture	73
3.4.3	Typical Running Sequence	76
3.5	Software Validation	78
3.5.1	Validation Methodology	78
3.5.2	Activity Simulator	78
3.6	Validation Results	80
3.7	Conclusion	82
3.8	Compléments sur le prototype Java	83
3.8.1	Données d'entrée de l'application	83
3.8.2	Illustration du processus de gestion des risques	86
	Conclusion	91
	A Détails sur l'utilisation du vérificateur de modèle Alloy	95
A.1	Présentation du modèle développé avec ALLOY	97
A.1.1	Structure du modèle	97
A.1.2	Comportement du modèle	101
A.1.3	Exécution de la vérification de modèle	103
A.2	Liste des propriétés ALLOY	104
A.2.1	Exemple de propriété	104
A.2.2	Liste complète des propriétés établies	105

TABLE DES MATIÈRES

B	Détails sur le prototype d'Inovus validant de l'approche de développement	107
B.1	Spécificités liées à l'utilisation d'OSGi	109
B.1.1	Exécution d'un bundle OSGi	109
B.1.2	Définition d'un service	111
B.2	Description des données d'entrée	112
B.2.1	Représentation de l'environnement	112
B.2.2	Description de situation à risque	114

Liste des figures

2.1	The INOVUS architecture in the home environment.	44
2.2	Classical process with the proposed additional step.	45
2.3	Processus de vérification formelle d'un modèle avec ALLOY.	55
3.1	INOVUS architecture as home safety solution.	68
3.2	ALLOY formel verification process.	71
3.3	INOVUS layers architecture and the resulting data flow.	72
3.4	INOVUS components diagram.	74
3.5	The INOVUS monitoring sequencing.	77
3.6	A possible INOVUS reaction services sequencing.	77
3.7	Extrait du logger du simulateur d'AVQ.	87
3.8	Extrait du logger des dangers détectés par l'application INOVUS.	88
3.9	Extrait du logger de rétroactions de l'application INOVUS.	89
A.1	Diagramme de classe utilisé pour construire le modèle ALLOY.	97

LISTE DES FIGURES

Liste des tableaux

1.1 Synthèse des risques à domicile et des processus de gestion associés, tels que traités dans la littérature.	31
2.1 Risks considered in the first increment.	48
3.1 Management technologies for range-related threats.	65
3.2 Effective risk coverage in the INOVUS prototype.	81
B.1 Grammaire de description de l'environnement cible.	115
B.2 Grammaire de description des contextes de risque.	116

LISTE DES TABLEAUX

Liste des programmes

A.1	Code ALLOY de l'entité <i>DecisionEngine</i> .	98
A.2	Signature abstraite en ALLOY modélisant un élément de cuisson, accompagnées des signatures dérivées.	100
A.3	Prédicat ALLOY modélisant le passage pour un capteur de température de l'état froid à l'état défini comme chaud.	101
A.4	Squelette du fichier d'exécution utilisé.	103
A.5	Exemple de propriété ALLOY vérifiée par le modèle formel avec <i>Alloy Analyzer</i> .	104
B.1	Extrait du fichier <i>Activator.java</i> du bundle <i>Reaction Manager</i> .	110
B.2	Interface du service proposé par le bundle <i>Reaction Manager</i> .	111
B.3	Définition du service proposé par le bundle <i>Reaction Manager</i> .	111
B.4	Extrait de la représentation XML de l'appartement-test du laboratoire Domus.	112
B.5	Extrait de la représentation XML des contextes de risques.	114

LISTE DES PROGRAMMES

Introduction

Ce mémoire propose une approche de développement logiciel pour une technologie d'assistance intégrée à un habitat intelligent. Cette approche se base sur l'utilisation de méthodes formelles pour assurer la cohérence et la fiabilité de la solution logicielle produite. Cette approche a été mise en pratique au sein du projet INOVUS du laboratoire Domus, dans le but d'assurer à l'utilisateur une utilisation sécuritaire de sa cuisinière.

Afin de positionner les travaux exposés dans ce mémoire, ce chapitre d'introduction présente le contexte de travail en terme de maintien à domicile ainsi que les solutions actuellement mises en oeuvre dans ce secteur, tout d'abord d'une manière globale, puis plus spécifiquement au laboratoire Domus, et enfin dans le cadre du projet INOVUS. Cela permettra ensuite la présentation des travaux de ce mémoire.

Le maintien à domicile : contexte de travail

Vieillesse et déficits cognitifs

Le vieillissement naturel chez l'être humain conduit à l'altération progressive des facultés physiques et cérébrales de la personne. Cette détérioration est source de difficultés, pour le sujet, à maintenir son rythme et ses conditions de vie habituelles au quotidien. Cette perte d'autonomie, caractérisée par l'apparition de déficits cognitifs chez le sujet âgé, s'accroît peu à peu avec le temps, ce qui mène fréquemment à son placement en institution spécialisée. Cependant, dans le cas de déficits cognitifs légers (e.g. dans les premières phases du vieillissement), le placement en institution peut être évité, au profit du maintien à domicile.

Dans une société vieillissante comme la nôtre, les personnes âgées sont majoritairement représentées au sein des sujets atteints de déficits cognitifs, néanmoins cette population intègre également les personnes souffrant de troubles de la santé, tels que les Traumatismes Cranio-Cérébraux (TCC), la Démence de Type Alzheimer (DTA), ou encore la schizophrénie.

Dans le cadre du maintien à domicile, ces personnes constituent une population d'individus qui, selon leur âge ou le degré de leurs déficits, nécessitent une assistance plus ou moins importante lors de la réalisation de leurs tâches de vie. Cette assistance est d'autant plus essentielle lorsque la perte d'autonomie concerne des Activités de la Vie Quotidienne (AVQ) telle que la toilette, les déplacements, ou bien encore les activités de cuisine [40].

Actuellement, cette assistance est principalement fournie par l'intervention d'aidants. En effet, pour permettre à la personne de conserver un minimum son rythme de vie et jouir de son cadre habituel malgré ses déficits cognitifs, le rôle des aidants s'impose afin de venir contrer les difficultés engendrées et permettre ainsi une meilleure qualité de vie pour la personne concernée. Ces intervenants extérieurs sont amenés à visiter plus ou moins régulièrement la personne, selon le degré de nécessité.

Cependant, avoir une personne déficiente à sa charge n'est pas une chose aisée, et cela tend fréquemment à compliquer le quotidien de l'aidant. Cet élément est mis en avant dans de nombreuses études qui parlent alors de *fardeau* - e.g. [26, 27, 66, 87] -, afin de faire ressortir l'ampleur des responsabilités qui retombent sur l'aidant. Thomas *et al.* [87] appuient tout particulièrement ce point en précisant que le rôle de l'aidant revient fréquemment à un membre proche du malade, et que la notion de *fardeau* est alors encore plus forte dès lors que l'aidant peut lui-même être sujet à certaines complications de son état de santé. Thomas *et al.* font alors référence à la *vulnérabilité* de l'aidant.

Apport des technologies d'assistance

La population atteinte de déficits cognitifs tend à s'accroître d'années en années, en accord avec l'augmentation de la moyenne d'âge de nos populations. Bharucha *et al.* [10] expliquent par exemple qu'aux États-Unis, la population des plus de 65 ans devraient doubler dans les 25 prochaines années pour atteindre les 72 millions, et que

INTRODUCTION

la proportion actuelle de personne souffrant de DTA (4.5 millions) devrait atteindre les 13 millions. Les États-Unis ne sont pas les seuls touchés, et une évolution similaire est à prévoir à l'échelle de la planète.

Pour accompagner ces personnes, il sera nécessaire d'adapter les processus d'assistance actuellement en place de manière à répondre adéquatement aux besoins de notre société. Les technologies d'assistance semblent toutes particulièrement désignées pour jouer ce rôle puisqu'elles constituent un élément clé dans le secteur du maintien à domicile en permettant d'alléger considérablement la charge des aidants, facilitant ainsi la vie au quotidien de la personne atteinte de déficits cognitifs.

Ainsi, de nombreux dispositifs sont actuellement élaborés par les chercheurs, souvent accompagnés par des industriels afin de permettre l'adoption de ces technologies par le grand public [10].

Les habitats intelligents

Les habitats intelligents constituent le second secteur particulièrement en expansion dans le domaine du maintien à domicile [16, 24]. En effet, le transfert des dispositifs d'assistance n'étant pas toujours aisé compte-tenu de la population ciblée, le but des habitats intelligents pour la santé est de venir intégrer directement les systèmes d'assistance au sein du domicile de la personne. Par le biais de la domotique, ces technologies peuvent alors être déployées de manière transparente pour l'utilisateur.

Le laboratoire Domus

Le laboratoire Domus¹ [25] est un laboratoire de recherche multi-disciplinaire dédié à la domotique et à l'informatique mobile. Il fut l'un des premiers acteurs dans le secteur de la recherche sur les habitats intelligents et du maintien à domicile. Les problématiques y sont variées et gravitent de manière générale autour de l'assistance cognitive. Les personnes âgées ou atteintes de déficits cognitifs (DTA, TCC, schizophrènes) constituent la population ciblée par les projets du laboratoire.

Convaincu que les technologies développées doivent être conçues en accord avec les

1. Laboratoire Domus - <http://domus.usherbrooke.ca/>

utilisateurs finaux, le laboratoire Domus accorde une importance toute particulière à la conception centrée utilisateur. Pour ce faire, les travaux qui y sont entrepris sont menés de concert avec des malades et leurs aidants, mais aussi avec des professionnels de la santé, de l'ergonomie, du design, ainsi que des industriels.

Les orthèses cognitives et les logiciels d'assistance à la réalisation de tâches représentent une part importante des travaux du laboratoire Domus, cependant des axes de recherche plus fondamentaux dans le cadre des habitats intelligents y sont abordés, tels que les réseaux de capteurs, la reconnaissance d'activité, la localisation intérieure/extérieure, la gestion dynamique de services au sein de l'habitat, ou encore la sécurité à domicile.

C'est ce dernier thème qui constitue le domaine de recherche abordé dans ce mémoire, au travers du projet INOVUS.

Inovus, la cuisinière intelligente

Le projet INOVUS est l'un des projets actuellement en cours de développement au laboratoire Domus. Ce projet a pour vocation de fournir une technologie d'assistance diffuse au sein de l'habitat intelligent, dans le cas d'une personne seule en maintien à domicile, et ce en observant en temps réel l'habitat afin prévenir et détecter l'apparition de dangers pour l'occupant lors de l'utilisation de la cuisinière.

On souhaite donc intégrer une certaine forme d'intelligence artificielle au sein de la cuisinière. C'est là l'origine du nom du projet INOVUS :

Intelligent Oven @ Université de Sherbrooke.

Thématique générale

Lors de la réalisation d'AVQ, les personnes âgées ou atteintes de déficits cognitifs peuvent faire face à des dangers pouvant conduire à un accident potentiellement fatal. Selon leur nature, les différentes pièces de l'habitation se trouvent plus ou moins sujettes à la présence de ces risques. Afin d'optimiser au mieux le maintien à domicile, il semble indéniable que des travaux nécessitent d'être menés dans ce domaine. Ce constat est d'autant plus renforcé lorsque l'on traite des habitats intelligents. En effet, parmi les nombreuses applications de l'informatique diffuse en domotique, la sécurité

INTRODUCTION

de la personne à son domicile constitue un élément inévitable : rendre intelligent le lieu de vie, outre les divers services qui peuvent être proposés, doit passer par l'assurance de la sécurité de l'occupant.

C'est sur ce terrain que vient se placer le projet INOVUS, en s'attachant plus particulièrement à l'utilisation de la cuisinière.

Motivation

Lors de la réalisation d'AVQ, certaines tâches peuvent menacer la sécurité de la personne dans l'habitat. Comme l'expliquera l'état de l'art présenté au chapitre 1, de nombreuses études sur les risques à domicile montrent qu'après la salle de bain, la cuisine constitue la seconde pièce dans laquelle surviennent la majorité des accidents domestiques. Ces accidents sont fréquemment dus à l'absence d'assistance lors de la réalisation d'AVQ *critiques*, dans le sens où elles peuvent menacer la sécurité de l'utilisateur. En effet, compte-tenu des complications engendrées par l'apparition et l'évolution des déficits cognitifs, la personne peut éprouver des difficultés à juger convenablement des dangers liés à ses actions, et l'accident peut alors survenir à tout instant. De cette manière, des tâches présentant initialement peu de dangers peuvent alors menacer directement la santé physique de l'occupant de l'habitat.

C'est sur ce constat que se base la motivation du projet INOVUS.

Enjeux et problématiques soulevées

Le projet INOVUS se concentre sur la détection et la prévention des risques physiques pour la personne, lors de la réalisation de tâches utilisant la cuisinière de l'habitat. Le but est de détecter, et dans la mesure du possible prévenir, l'apparition de risques pour la personne lors de l'utilisation de la cuisinière. Il est donc important de connaître les dangers identifiés au sein de l'habitat, pour ensuite définir les risques à prendre en considération au sein du projet INOVUS. L'identification des situations dangereuses à traiter et de leurs contextes d'apparition correspond donc au premier défi de ce projet.

Lorsque les risques à traiter sont connus, il s'avère ensuite nécessaire de déterminer par quels moyens nous allons être en mesure de les détecter, puis de les traiter.

Le projet INOVUS se base sur le postulat que le système produit sera installé dans un habitat intelligent. Les technologies qui y sont présentes doivent donc non seulement permettre de surveiller les paramètres constituant chacune des situations dangereuses, mais aussi de fournir des réactions appropriées face à un danger auquel l'utilisateur n'aurait pas prêté attention. Il s'agit donc ici de la caractérisation des risques en fonction des ressources présentes dans l'environnement et des informations mises à disposition. C'est cette caractérisation qui va nous permettre de mettre au point les algorithmes de détection de chacun de risques que l'on souhaite traiter.

Enfin, la dernière problématique à laquelle il nous faut répondre concerne la rétroaction à fournir, aussi bien à l'utilisateur que sur le système. En effet, le but du projet INOVUS ne se contente pas simplement de la détection de situation dangereuse ; les exigences du projet en matière de sécurité de la personne imposent naturellement de fournir une réaction systématique. Ainsi, il a été choisi de fournir dans un premier temps une information à l'utilisateur concernant l'apparition d'une situation dangereuse. Les capteurs présents dans l'environnement permettront alors de s'assurer de la bonne réception de l'information de la part de l'utilisateur. Dans le cas où ce dernier ne prendrait pas en considération une situation dangereuse pour laquelle il a été préalablement informé, on doit être en mesure d'agir sur les sources du danger afin d'écartier l'accident domestique. De là se pose donc la question des moyens utilisables pour venir interagir directement dans l'environnement. C'est donc ce qui constitue le troisième et dernier enjeu du projet INOVUS.

Multidisciplinarité du projet

Les problématiques soulevées au paragraphe précédent mettent en exergue l'aspect multidisciplinaire du projet INOVUS. En effet, de par sa nature et son contexte, ce projet fait appel à des domaines de recherches variés, technologiques certes, mais aussi cliniques (psychologie, ergothérapie, ergonomie, gérontologie).

En particulier, afin d'informer un utilisateur de la présence d'un danger, il est nécessaire de prendre en compte ses déficits pour pouvoir s'assurer de la bonne réception et compréhension du message envoyé. Il faut donc adapter les modes d'interactions homme-machine de manière à fournir une information qui soit aisément interprétable par l'utilisateur. Ces considérations relèvent plus alors des domaines de l'ergonomie et de

INTRODUCTION

l'ergothérapie, et n'auront donc pas été traitées dans ces travaux de maîtrise.

D'un point de vue technologique, le projet INOVUS combine principalement l'électronique en ce qui concerne l'acquisition de données, et l'informatique en ce qui concerne le traitement des données recueillies.

Aspect matériel Dans la mesure où les travaux de maîtrise présentés dans ce mémoire ne portent que sur la couche logicielle du projet, l'acquisition physique des données de l'environnement servant à détecter les contextes à risques n'a pas été traitée. En fait, il s'agit là des travaux d'un autre étudiant à la maîtrise au laboratoire Domus - Thomas Tessier -, qui se charge de déterminer les facteurs mesurables constituant une situation dangereuse, et les moyens de les capter depuis l'environnement à partir de senseurs spécifiquement sélectionnés.

Pour la couche logicielle présentée dans ce mémoire, les données d'entrée utilisées ont été fournies par un simulateur d'AVQ existant au laboratoire [14]. Ce simulateur permet de reproduire le déroulement d'un scénario de vie au sein d'un habitat, et permet de générer les sorties capteurs correspondantes.

Aspect logiciel D'un point de vue applicatif, les exigences du projet INOVUS sont de fournir un logiciel qui soit en mesure de traiter des données d'entrée représentant l'environnement de l'habitat - plus spécifiquement la cuisine -, et de les interpréter. Le traitement apporté à ces données doit permettre d'identifier les configurations de l'environnement présentant des dangers pour son occupant. Le système devrait également être en mesure de détecter directement dans l'environnement les actions de l'utilisateur permettant de venir inhiber la présence de dangers. Au besoin, le système doit pouvoir se substituer à l'utilisateur en actionnant les dispositifs permettant d'éliminer les sources de dangers (ici donc, les éléments de cuisson). De plus, lorsqu'un risque persiste, on doit ensuite être en mesure d'interagir avec l'utilisateur au travers des actionneurs présents dans l'habitat. Enfin, il serait utile aussi d'être en mesure de communiquer avec des intervenants extérieurs, aidants ou services d'urgence, et ce de manière automatique lorsqu'un risque critique se présente.

Contribution scientifique du projet Inovus

Le projet INOVUS se veut être une solution globale aux dispositifs de détection et de prévention des dangers pour l'occupant de l'habitat lors de la réalisation de tâches de cuisine. Comme le montrera la revue de littérature au chapitre 1, les solutions actuelles ne répondent de manière générale qu'à un risque ponctuel, et il est alors nécessaire de combiner plusieurs technologies pour parvenir au même degré de couverture. Cette combinaison de dispositifs peut alors s'avérer difficile à installer et maintenir. Le projet INOVUS souhaite venir pallier à ce problème en proposant, à partir d'un habitat intelligent, une solution complète, modulable, et aisément paramétrable en fonction des situations de risques que l'on souhaite couvrir.

Problématique de M.Sc.

Le contexte de travail présenté précédemment a permis de positionner le projet INOVUS et l'intérêt qu'il représente. Ce projet a fait ressortir deux axes de recherches majeurs, sur les plans matériels, et logiciels. La problématique de recherche de cette maîtrise concerne la proposition d'une approche de développement adaptée pour les habitats intelligents, et qui permette de fournir une réponse satisfaisante à la couche logicielle du projet-INOVUS.

Objectifs et proposition

Le développement de technologies d'assistance dans les habitats intelligents entraîne une forte criticité des systèmes, dès lors que les projets comportent des enjeux humains. Pour cette raison, la solution produite doit être en mesure d'être opérationnelle en tout temps. Cela implique alors des contraintes fortes en termes de fiabilité, modularité, et contextualisation.

La finalité de ces travaux de maîtrise est donc de venir proposer une approche de développement logiciel qui réponde du mieux possible à ces critères et puisse ainsi tenter d'apporter une réponse spécifiquement adaptée aux habitats intelligents. Afin de valider cette approche, un prototype logiciel a été réalisé dans le cadre du projet-INOVUS. Ce prototype permet de plus de réaliser une preuve de concept d'une telle

INTRODUCTION

solution de sécurité.

L'une des exigences critiques du projet INOVUS étant de fournir une application fiable aussi bien en termes de détection de risques que de réaction du système, la proposition de ces travaux est de venir intégrer l'utilisation de méthodes formelles dans le cadre d'INOVUS. Un modèle de solution pour la couche logicielle de ce projet a donc été construit, puis une spécification formelle y a été appliquée. Cette spécification formelle s'est basée sur le vérificateur de modèle ALLOY² [53]. L'utilisation d'un tel outil permet d'assurer d'un point de vue mathématique la cohérence ainsi que le bon comportement de l'application développée, et ce dès sa conception.

D'autre part, comme mentionné précédemment, la modularité et la contextualisation du système produit étant des pivots fondamentaux de l'application à fournir, la conception et l'implémentation ont été réalisées de manière à permettre aisément aussi bien l'adaptation des contextes de risques déjà intégrés que les algorithmes de raisonnements. Cela a été permis notamment grâce à l'utilisation de patrons de conception.

Enfin, afin d'intégrer au mieux le système INOVUS au sein d'un habitat intelligent, l'implémentation a été réalisée à l'aide du cadriciel *OSGi*³ [6]. *OSGi* est une plateforme dynamique de gestion de composants spécialement développée pour le langage Java, et dont les spécifications se trouvent particulièrement adaptée à une utilisation en informatique diffuse. L'utilisation d'une telle technologie permet donc d'intégrer parfaitement le système INOVUS au sein des services actuellement disponible dans l'habitat intelligent du laboratoire Domus.

D'autre part, l'utilisation de ce cadriciel permet d'isoler clairement les composants à paramétrer selon l'utilisation souhaitée, sans impact sur le reste du système. Ce point vient donc là encore répondre aux contraintes de modularité et contextualisation imposée par le projet INOVUS.

Méthodologie

Les travaux de M.Sc. présentés dans ce mémoire se sont décomposés en trois étapes majeures : revue de la littérature sur les solutions de sécurité dans la cuisine et

2. Alloy Analyzer 4 - <http://alloy.mit.edu/alloy4/>

3. Open Services Gateway initiative - <http://www.osgi.org/>

l'utilisation des méthodes formelles dans les habitats intelligents, construction et vérification d'un modèle de solution, puis validation de cette approche de développement par le biais du développement d'un prototype fonctionnel pour le projet INOVUS.

Revue de la littérature

Tout d'abord, un travail de recension des risques identifiés au sein de l'habitat a dû être effectué, et ce afin de déterminer les dangers les plus à même d'apparaître lors de la réalisation de tâches liées à l'utilisation de la cuisinière. D'autre part, afin de justifier le concept de solution de sécurité proposé, il a été nécessaire de faire l'état de l'art des technologies existantes, en lien avec la sécurité de la personne lors de l'utilisation de la cuisinière de l'habitat. Enfin, le troisième volet de ce jalon correspond à la revue des utilisations des méthodes formelles de spécification dans les habitats intelligents.

Construction et vérification du modèle de solution

À partir de la revue de littérature sur les risques à domicile et leurs solutions actuelles, un modèle de solution à la problématique du projet INOVUS a été élaboré. Quelques situations de risques ont été sélectionnées pour constituer un échantillon de contextes types, et ainsi servir de support à ce processus. Le raisonnement afférent à la détection de ces contextes et aux réactions à fournir en conséquence a été réduit au plus simple.

C'est à ce niveau que la phase de spécification formelle a été insérée, afin de s'assurer que la solution logicielle envisagée permette bien de rencontrer les exigences du projet. Un modèle ALLOY représentant la solution envisagée a donc été construit, et vérifié par le biais des outils fournis avec ce vérificateur de modèle.

Développement du prototype pour valider l'approche proposée

En se basant sur le modèle développé et vérifié lors de la spécification formelle de la solution proposée, une implémentation en Java a été réalisée. L'objectif de cette implémentation est de fournir un prototype fonctionnel de l'application INOVUS, basé sur l'approche de développement proposée.

INTRODUCTION

L'utilisation du simulateur d'AVQ du laboratoire Domus a permis de valider ce prototype : à partir des données de simulation concernant la réalisation de scénarios de vie courante dans l'habitat, les situations à risques initialement sélectionnées sont détectées avec succès, et les réactions qui en découlent sont bien appliquées.

Ce prototype fonctionnel constitue à la fois une preuve de concept de la solution de sécurité envisagée, et conduit dans le même temps à la validation de l'approche de développement proposée.

Contribution scientifique des travaux de M.Sc.

Les travaux présentés dans ce mémoire viennent proposer l'utilisation de la vérification de modèle (*model checking*) dans le cadre des habitats intelligents. D'une manière générale, la littérature ne fait état que de très peu d'utilisations des spécifications formelles dans ce contexte. En ce qui concerne l'utilisation du vérificateur de modèle, en date de publication du mémoire, seuls les travaux de Biswas *et al.* [11] ont pu être référencés. Les travaux présentés dans ce document innovent donc en venant introduire l'utilisation d'un tel outil pour le développement de technologies d'assistance dans les habitats intelligents.

D'autre part, la mise en application de l'approche de développement proposée a permis d'aboutir à un prototype d'application permettant de gérer un ensemble de données représentatives d'un environnement, afin d'en extraire des situations à risques pré-identifiées. Cela constitue une première réponse à la problématique logicielle soulevée par le projet INOVUS, et représente ainsi un apport non négligeable au sein d'un projet innovant et au potentiel de développement très élevé.

Le format choisi pour ce mémoire est par insertion d'articles. La revue de littérature précédemment évoquée sera donc accompagnée des deux publications actuelles concernant la couche logicielle du projet INOVUS. Ce choix est motivé par le fait que ces deux papiers ont été rédigés par l'étudiant au cours de ses travaux de maîtrise.

De cette manière donc, le chapitre 1 présente l'état de l'art associé au projet. Le

INTRODUCTION

chapitre 2 présente quant à lui l'approche employée au cours du développement de ce prototype, et ce au travers d'un article publié au sein de la conférence PerCom 2011⁴ [1]. L'article de journal⁵ du chapitre 3 présente la mise en application, dans le cadre du projet INOVUS, de l'approche de développement proposée. Enfin, un chapitre de conclusion viendra clore ce mémoire en faisant un bilan des résultats proposés et en amenant des axes de recherches futures.

4. PerCom 2011 - <http://www.percom.org/2011/>

5. Non soumis - Choix du journal à définir.

Chapitre 1

Revue de la littérature

Nous identifierons dans la première section les risques présents d'une manière générale au domicile de l'occupant. Les solutions actuelles qui y sont associées sont ensuite présentées dans la seconde section.

La troisième section de cet état de l'art propose une synthèse des deux sections précédentes. Cela permet alors de positionner clairement le projet INOVUS, et ainsi justifier l'apport des travaux présentés dans ce mémoire.

Enfin, la solution logicielle développée pour le projet INOVUS fait appel aux spécifications formelles afin d'assurer l'efficacité et la fiabilité du logiciel final. Afin de justifier cette approche, la dernière section traite donc de l'utilisation des méthodes formelles dans les habitats intelligents.

1.1 Généralités sur les risques à domicile

1.1.1 Entrée en matière

Les risques à domicile constituent un domaine de recherche relativement bien couvert par la communauté scientifique. En particulier, depuis la fin des années 1990 les études cliniques sur le sujet ont commencé à se multiplier. Néanmoins, une revue de la littérature actuelle fait apparaître le fait que les risques à domiciles ne sont pas traités directement. En effet, les études menées portent pour la plupart sur les accidents domestiques. Ces études s'intéressent donc plus à l'aspect *accident*, plutôt qu'à la notion de *danger*. Ce point peut sembler anodin, néanmoins selon l'approche adoptée la portée des contributions ainsi que leur finalité sont alors très différentes.

D'autre part, il est important de préciser que ces études ont pour la majorité trait aux domaines de la psychologie, de la gérontologie, ou encore de l'ergothérapie. Les contributions à nature technologique s'avèrent de ce fait restreintes.

Ce n'est que depuis le développement récent des *gérontechnologies* et des habitats intelligents que l'intérêt s'est porté sur les causes des accidents domestiques et non plus uniquement sur leurs conséquences. Cet essor est d'ailleurs d'autant plus justifié par la démocratisation du maintien à domicile au sein de notre société.

1.1.2 Définitions

Nombre d'études sur les accidents domestiques s'avèrent en fait être des évaluations menées sur une population de personnes volontaires, directement au sein de leur habitation, afin d'identifier les facteurs de risques chez la personne atteinte de déficits cognitifs (e.g. [15, 40, 85]).

On remarque que selon les auteurs, la définition des risques n'est pas la même. Certains intègrent uniquement les risques directement liés à l'environnement, alors que d'autres considèrent comme risque tout ce qui découle des déficits cognitifs et qui peut porter préjudice à la personne, aussi bien d'un point de vue physique que d'un point de vue moral. Il semble donc pertinent de préciser la signification de quelques termes.

Ainsi, dans les articles étudiés, les notions de *risque* et *facteur de risque* sont sou-

1.1. GÉNÉRALITÉS SUR LES RISQUES À DOMICILE

vent considérées comme identiques. Seule la notion d'*accident* est commune chez tous les auteurs, et correspond d'une manière générale aux conséquences de la concrétisation d'un risque dans un contexte particulier. Cette confusion syntaxique a pour conséquence de limiter très fortement la distinction entre le risque à part entière et sa source [15].

Pour illustrer ce constat, prenons l'exemple de la chute d'une personne. Selon les auteurs, l'élément qui aura causé la chute sera considéré comme étant lui-même le risque (e.g. *trébucher sur un tapis*), alors que d'autres le considéreront comme le contexte de concrétisation du risque (le risque étant la chute).

Dans le cadre de ce mémoire, nous distinguerons les notions de *risque*, *contexte d'apparition* et *accident* de la manière suivante : un *risque* regroupe l'ensemble des situations qui peuvent conduire à un accident de même nature, le *contexte d'apparition* étant l'ensemble des facteurs qui ont menés à la concrétisation de ce risque ; un *contexte d'apparition* correspond donc à l'une des situations menant à un *risque* ; l'*accident* représente quant à lui la conséquence directe de cette concrétisation.

1.1.3 Risques identifiés dans la littérature

Peu d'études généralistes ont été menés concernant les risques à domicile. Outre quelques publications présentant des outils d'évaluation de la sécurité à domicile (cf. 1.2.2), les travaux recensés ne se concentrent que sur un risque en particulier.

Dans le cadre de sa thèse de doctorat [12] traitant des situations des risques à domicile, Bourgeois a conduit une revue des dangers majeurs pouvant menacer la santé physique de l'occupant souffrant de démence.

Ainsi, la multitude de contextes de risques identifiables dans l'habitat peut être organisée parmi les menaces majeures suivantes : la malnutrition, la médication, les incendies, les brûlures, les chutes, les intoxications, et enfin l'errance.

On remarque que ces dangers sont de nature très différentes, il est donc nécessaire de les étudier individuellement pour être en mesure de déterminer les contextes de risques qui vont être pertinents pour le projet INOVUS.

Risques majeurs

Les risques considérés ici sont ceux qui présentent des conséquences pouvant s'avérer fatales pour l'utilisateur. Ils sont donc étroitement liés aux contextes environnementaux, la vulnérabilité des personnes âgées ou souffrantes s'en trouve alors accrue.

Chutes Les chutes constituent l'accident le plus fréquent chez les personnes de plus de 75 ans selon Bourgeois *et al.* [13], et parmi les dix premières causes de mortalité accidentelle. Pour cette raison, il est le thème le plus largement abordé dans la littérature sur les risques à domicile (e.g. [62, 21, 39, 22, 88, 54, 70]).

Chez les personnes âgées l'affaiblissement physique lié au vieillissement tend à rendre la personne plus vulnérable à son environnement. Des études telles que [56, 79] précisent qu'un environnement inadapté à l'état de santé de la personne constitue un facteur aggravant. Puisieux *et al.* [79] indiquent de plus que les chutes et les troubles cognitifs sont étroitement liés chez le sujet âgé.

En terme de conséquences, la chute est la première cause de placement en institution d'après Bourgeois *et al.* [13].

Brûlures Que l'on se place dans le cadre du domicile ou en dehors, la littérature fait état de types de brûlures variés. En effet, qu'elles soient de nature électriques, par contacts (surface ou liquide à température élevée), par échauffement (frottements ou solaire), ou encore chimiques, les brûlures représentent un danger omniprésent dans nos activités de la vie quotidienne.

Ainsi, dans l'habitat, les brûlures constituent la seconde menace la plus importante pour l'occupant. Ehrlich *et al.* [29] expliquent ce constat par les nombreuses sources qui peuvent en être la cause. Par exemple, des brûlures peuvent survenir aussi bien en utilisant de l'eau chaude, qu'en branchant un appareil électrique sur une prise, en manipulant des produits ménagers, ou encore en déplaçant un ustensile de cuisine sur un rond de la cuisinière en utilisation.

De ce constat découlent d'ailleurs nombre de travaux, principalement à vocation statistique, afin d'identifier au mieux les causes des brûlures en fonction du type de population concernée.

Citons notamment Hill *et al.* [47], qui a consigné l'ensemble des admissions de pa-

1.1. GÉNÉRALITÉS SUR LES RISQUES À DOMICILE

tients de plus de 65 ans à l'hôpital de Vancouver suite à des accidents domestiques menant à des brûlures, entre 1985 et 1999.

En ce qui concerne les causes de ces blessures, les contextes identifiés sont l'utilisation d'eau chaude, de radiateurs, de cheminées, ainsi que de fours ou plaques de cuisson. En terme de conséquences, cette étude montre que la surface moyenne brûlée représente 8.9% du corps. La nature des brûlures est répartie de la sorte : brûlures électriques (3%), par échauffement (41%), et par le feu (53%). Le taux de mortalité en regard de ces blessures est de 58.9% pour une hospitalisation moyenne de 33 jours, ce qui indique des soins longs, importants, et pour lesquels le pronostic vital de la personne est engagé. D'autre part, trois facteurs augmentant les risques sont mis en évidence : l'alcool (21.4%), la cigarette (10%), et la démence (33%). Autres chiffres intéressants, 52% des brûlures par échauffement viennent de la toilette, avec 66% de personnes démentes et un taux de mortalité de 41%.

On constate alors que la démence constitue réellement un facteur aggravant du niveau des risques de brûlures chez les personnes âgées de plus de 65 ans.

En 2005, Ehrlich *et al.* [30] s'étaient déjà intéressés aux brûlures mineures traitées sur des personnes âgées de plus de 65 ans dans un service d'urgence de la ville de New York pendant 6 ans. Les résultats de leur étude font ressortir l'éthiologie des brûlures dans l'habitat en fonction de leur type, ainsi que la répartition des accidents dans les différentes pièces. La cuisine est alors largement majoritaire, avec 65% des occurrences d'accidents. La salle de bain, qui positionne les chutes en tant que premier risque dans l'habitat, se trouve alors en deuxième position avec 13% des accidents enregistrés.

Ces études permettent alors de conclure que les brûlures liées aux AVQ réalisées dans la cuisine représentent un risque majeur.

Incendies Là encore, s'agissant d'un risque bien connu et dont les conséquences peuvent s'avérer fatales, les incendies sont relativement bien couverts dans la littérature actuelle (e.g. [51, 55, 65, 91]). Les études menées présentent de nombreuses statistiques quant aux accidents domestiques qui y sont liés.

Néanmoins, les travaux proposés sont souvent destinés à identifier les causes des incendies domestiques et leurs conséquences sans s'attacher à une population spécifique. Pour cette raison, la vulnérabilité aux accidents par le feu des personnes âgées

CHAPITRE 1. REVUE DE LA LITTÉRATURE

ou démentes apparaît dans certains travaux, mais aucuns ne traitent explicitement des incendies dans les populations souffrant de déficits cognitifs [13].

Cependant, en 1999, Killalea [55] présente une revue des études concernant les facteurs de déclenchements d'incendies dans l'habitat, menées entre 1990 et 1998. Il en ressort que les sources d'incendies domestiques sont très dépendantes du type de population au sein du logement. Les enfants, les personnes âgées, infirmes, ou nécessitant d'importantes médications sont alors identifiés comme étant les plus vulnérables. Ces populations sont d'autant plus menacées dès lors qu'elles combinent d'autres facteurs aggravants tels que la cigarette (e.g. assoupissement avec une cigarette allumée) [65, 91].

D'autre part, en 2006, Hall [43] propose une étude détaillée des contextes favorables aux incendies lors de la réalisation de tâches de cuisine. Les incendies internes au four se trouvent être la principale cause des départs de feu dans la cuisine de l'habitat, suivi par l'utilisation sans surveillance des ronds de cuisson, et enfin les dispositifs portables tels que le toaster.

De manière générale, nous pouvons donc retenir les tâches de cuisine, l'utilisation du fer à repasser, un foyer actif dans la cheminée, ou encore la cigarette, comme étant les contextes propices aux départs de feu identifiés dans la littérature.

Intoxications par les gaz et fumées Les intoxications par inhalation sont relativement peu traitées dans la littérature. Seuls les accidents liés au monoxyde de carbone bénéficient de nombreux travaux (e.g. [78]), mais il ne s'agit pas là des seules causes d'intoxication dans l'habitat. Ainsi, Bourgeois *et al.* [13] en identifient deux autres sources lors du maintien à domicile : les fumées associées aux incendies, ainsi que la manipulation de produits d'entretien ménager.

La majorité des travaux sur les incendies mentionnent dans le même temps les accidents respiratoires qui en découlent (e.g. [55]), mais il est rare qu'une étude approfondie y soit associée.

Quelques auteurs en toxicologie prêtent aussi attention aux intoxications par les produits ménagers. Baudet *et al.* [9] fait d'ailleurs ressortir le fait que ce type d'accidents sont majoritairement liés à l'ingestion des produits, plus qu'à leur inhalation. Ce type d'accidents est généralement causé par une incapacité à lire les étiquettes

1.1. GÉNÉRALITÉS SUR LES RISQUES À DOMICILE

des contenants, ce qui est alors propice à des confusions avec des produits digestes ou avec une manipulation inappropriée.

Baudet *et al.* expliquent de plus que d'importants efforts sont apportés pour éviter ces situations, mais que la solution la plus indiquée reste la prévention par l'information et la sensibilisation aux dangers liés à la manipulation de tels liquides.

D'autre part, des études concernant les incendies identifient les intoxications par la fumée comme étant une des conséquences liées à un départ de feu [5]. Ce point constitue donc le seul contexte d'apparition que nous retiendrons en ce qui concerne les intoxications dans le cadre du projet INOVUS, en particulier dans le cadre d'un dégagement de fumées incontrôlé lors d'activités de cuisson.

Risques additionnels

On s'intéresse dans cette sous-section aux dangers n'ayant pas d'incidence physique instantanée, ou n'entraînant pas d'hospitalisation immédiate. Par leur nature, ces risques sont intrinsèquement écartés du projet INOVUS, leurs contextes d'apparition étant relativement éloignés de l'utilisation de la cuisinière.

Médication D'après [13], il est statistiquement prouvé qu'au-delà de 65 ans, la médication quotidienne est quasi-permanente. De ce fait, les personnes âgées sont alors facilement confrontées à des difficultés à suivre leurs prescriptions, aussi bien en terme de type de médication que d'horaire ou de posologie. Ces complications peuvent en particulier être causées par des étiquettes difficilement lisibles ou des emballages portant à confusion. Le risque d'erreur médicamenteuse est alors très élevé et peut conduire à une intoxication menant à l'hospitalisation, voire au décès du sujet dans les cas les plus graves.

Alimentation Des études, dont celles de Watson *et al.* [92], montrent les conséquences de la démence sur les capacités de la personne à se nourrir, et les effets néfastes qui peuvent en découler. Ainsi, [13] recense une série de travaux portant sur les habitudes nutritionnelles chez les personnes âgées ou atteintes de déficits cognitifs. Perte de poids, dénutrition, malnutrition et intoxication alimentaire sont présentés comme étant les problèmes les plus fréquents.

CHAPITRE 1. REVUE DE LA LITTÉRATURE

Outre l'intoxication alimentaire, ces éléments ne présentent pas de dangers à très court terme pour la personne. Néanmoins, il est intéressant de préciser que ces problématiques ont été adressées par plusieurs auteurs, qui proposent alors des solutions technologiques pour réguler ou monitorer les aliments consommés, permettant ainsi le suivi par les aidants et favorisant une alimentation équilibrée [18, 20].

Errance et désorientation L'errance correspond à une instabilité psychomotrice qui conduit la personne à se déplacer dans l'habitat sans but précis. La désorientation, quant à elle, correspond à la perte des repères spatiaux.

Il s'agit des troubles du comportement liés aux déplacements dans l'habitat. Bien qu'étant identifiés dans la littérature, ils ne présentent pas de dangers physiques pour la personne. Toutefois, des études telles que celles menées par Rolland *et al.* [81] ou encore Tilly *et al.* [88] montrent que ces troubles peuvent tout de même engendrer des blessures lorsqu'ils conduisent à la chute de la personne déficiente.

Gestion budgétaire Il s'agit ici des difficultés de la personne à gérer ses finances personnelles. Cet élément sort du cadre de notre étude, mais certains auteurs tels que Gilmour [40] le considèrent comme un risque à domicile. Il ne s'avère pas pertinent ici puisqu'il ne constitue pas un danger sur le plan physique, il est donc écarté de la suite de cet état de l'art.

1.2 Gestion des risques à domicile

La section précédente a mis en évidence les risques pour la personne, identifiés au sein de l'habitat. On peut ainsi remarquer que ces risques sont nombreux et qu'il peut, comme le montrent Lefebvre *et al.* [56], être difficile d'y palier. Cette section a pour but de présenter les moyens de gestions des risques à domicile offerts.

La majeure partie des solutions proposées s'avère en fait être des conseils ou des bonnes pratiques à mettre en œuvre. Toutefois, de nombreuses études, ayant pour objectif l'identification et l'évaluation précises des facteurs de dangers potentiels pour la personne lors de la réalisation d'AVQ, ont été menées. Certaines seront donc

1.2. GESTION DES RISQUES À DOMICILE

présentées ici, puis seront suivies des technologies identifiées pour traiter les risques à domicile les plus importants.

Les sections suivantes synthétiseront ensuite cette revue sur les risques à domicile, puis la restreindront à la problématique du projet INOVUS concernant la cuisinière.

1.2.1 Prévention et sensibilisation

Les risques à domicile concernent tous types de populations, néanmoins la section précédente a fait ressortir le fait que certaines personnes y sont plus vulnérables que d'autres. Pour cette raison, de nombreux organismes (e.g. l'*Alzheimer's Society*¹ [83]), ou compagnies d'assurances (e.g. l'assureur français MAIF² [63]) proposent des pamphlets ou des sites internet au travers desquels sont listés les conseils qu'ils diffusent en terme de bonnes pratiques. Deux objectifs principaux peuvent être identifiés dans cette démarche. D'une part, il s'agit d'adapter l'environnement de la personne concernée et réduire ainsi les facteurs qui engendrent l'apparition de risques. D'autre part, cela permet de sensibiliser la personne elle-même et ses aidants aux dangers qui l'entourent.

Il est important de noter que dans le cadre des personnes âgées ou atteintes de déficits cognitifs, les risques pour la personne sont en grande partie dus à un environnement inadapté. Une intervention au niveau de l'aménagement de l'habitat mène de manière automatique à une réduction des contraintes pour la personne déficiente, et donc naturellement à une réduction des risques auxquels elle est sujette.

C'est dans cette optique que sont menés de nombreux travaux en ergothérapie, comme décrit dans la sous-section suivante.

1.2.2 Évaluations de l'habitat

De nombreuses études destinées à identifier les éléments de l'habitat représentant des facteurs de risques pour les personnes souffrant de déficiences peuvent être recensées dans la littérature. Ces travaux ont pour la plupart été conduits avec les malades et leurs aidants.

1. <http://www.alzheimers.org.uk/>

2. <http://www.le-maintien-a-domicile.fr/>

CHAPITRE 1. REVUE DE LA LITTÉRATURE

En particulier, Anemaet et al. [7] proposent un examen précis des éléments composants l'environnement de vie de la personne, à la fois intérieur et extérieur, afin de consigner les zones pouvant potentiellement présenter des dangers pour la personne et suggérer les modifications à apporter pour palier à ces risques. Cette évaluation est réalisée dans l'objectif de rendre l'habitat plus sûr et/ou fonctionnel. Des standards pour l'évaluation des risques sont aussi proposés, tels que ceux de la *Joint Commission on Accreditation of Healthcare Organizations* [7].

De manière plus générale, la majorité des études dans le domaine des risques à domicile aboutissent à l'établissement de grilles d'évaluation. Ainsi, nombre de ces outils s'intéressent à l'aménagement intérieur du domicile, la qualité des sols, des installations électriques, ou bien encore plus simplement à l'ergonomie et l'optimisation des espaces de vie. Il ressort ensuite de ces études des scores représentatifs du niveau d'adaptation de l'environnement en regard des déficits de la personne, de manière à souligner les éventuelles lacunes dans chacun des secteurs critiques de la maison. Pour exemple, Anemaet *et al.* proposent l'utilisation du *Functional Environment Assessment*, de l'*Objective Safe at Home*, ainsi qu'une liste de vérifications concernant l'ensemble des points potentiellement critiques pour la personne.

Suivant le même objectif, Poulin de Courval [77] suggère quatre outils différents, selon la population à laquelle on s'adresse ou encore le type d'informations que l'on veut faire ressortir : Grille d'Évaluation de la Sécurité (GES [77]), *Safety Assessment of Function and the Environment for Rehabilitation* (SAFER Tool [57]), *Westmead Home Safety Assessment* (WeHSA [21]), *Home Falls and Accidents Screening Tool* (HOME FAST [62]).

Notons qu'il ne s'agit là que d'une proposition de l'auteur, et que d'autres grilles d'évaluations des dangers sont aussi disponibles (e.g. [70, 84]).

De plus, outre l'évaluation directe des dangers au domicile, des outils sont aussi proposés pour évaluer la prise de conscience des risques de la part des malades, mais aussi de la part des aidants. Ces outils sont principalement destinés à améliorer la connaissance des dangers de l'habitat en terme de nature et de contextes d'apparition (e.g. [80]). Ils ont donc plus une vocation éducative ou de sensibilisation.

1.2. GESTION DES RISQUES À DOMICILE

1.2.3 Technologies d'assistance

Les solutions technologiques destinées à accompagner les personnes atteintes de déficits cognitifs sont multiples et variées [10, 58, 41, 68]. Le panel ainsi constitué offre une couverture relativement large des risques à domicile, mais il subsiste malgré cela des lacunes. Parmi les domaines couverts, on retiendra en particulier les solutions de suivi de la personne, d'assistance à domicile, et plus récemment le développement des habitats intelligents.

Assistance au domicile

Les paragraphes ci-après décrivent les solutions technologiques qui existent actuellement pour prévenir ou détecter l'apparition des risques à domicile.

Parce qu'il existe une multitude de dispositifs d'assistance en ce qui concerne la réalisation d'AVQ, les déplacements, l'alimentation, ou encore la médication, seuls les risques majeurs présentés en 1.1.3 sont détaillés ici.

Incendies domestiques La technologie la plus répandue est le détecteur de fumée [5, 17, 51]. De nombreuses études, telles que celle menée par Mallonee *et al.* [65] ou encore Ahrens *et al.* [5], montrent que l'utilisation de tels boîtiers permet de réduire sensiblement le taux de blessures, dans la mesure où une vérification régulière de l'état du système est effectuée.

En effet, bien que cette technologie soit l'une des plus répandue, une fois installée au domicile de l'occupant ce dernier n'est pas toujours à même d'en assurer le maintien. Pour cette raison, malgré leur présence les détecteurs de fumées ne sont pas toujours fonctionnels. Cette situation se retrouve dans 18% des cas d'après Ehrlich *et al.* [29]. Selon Ahrens *et al.* [5], la batterie est identifiée comme étant la plus importante cause de défectuosité : dans 54% des cas elle est absente ou déconnectée, et dans 19% des cas elle n'est plus fonctionnelle ou déchargée.

Malgré son large déploiement, Ahrens *et al.* précisent de plus une série de facteurs limitant l'efficacité de cette technologie. Elle n'est donc pas parfaite et des progrès doivent y être apportés.

Il est toutefois important de préciser que l'important pourcentage de dysfonctionne-

CHAPITRE 1. REVUE DE LA LITTÉRATURE

ments liés à une batterie absente ou déconnectée s'avère dans certains cas être un choix délibéré de la part de l'utilisateur, conséquemment aux fausses alarmes qui peuvent se déclencher de manière intempestive.

Brûlures Nous avons introduit en 1.1.3 les risques de brûlures dans l'habitat. Ces risques sont très peu traités d'un point de vue technologique.

Ainsi, seules les brûlures par l'eau chaude bénéficient de dispositifs de prévention. En effet, les limiteurs de température sont largement répandus³ [37]. Certains produits intègrent de plus directement sur les robinets des dispositifs lumineux représentatifs de la température de l'eau (e.g. Robi-Light⁴ [38], High Tech Place⁵ [76] et Luisina⁶ [61]).

Toutefois, en ce qui concerne les surfaces chaudes, la majorité des cuisinières intègrent des indicateurs lumineux, ce qui permet d'informer l'utilisateur du danger. Malheureusement, dans le cas des populations ciblées par les travaux de recherche de ce mémoire, il est fréquent que l'utilisateur ne perçoivent pas du tout, ou de manière incorrecte cette information. Cela montre que cette technologie n'est pas adaptée à tout type d'utilisateurs de la cuisinière.

Chutes Deux catégories de technologies ont été identifiées pour détecter les chutes à l'intérieur du domicile.

D'une part, il existe de nombreux bracelets ou pendentifs permettant d'alerter les aidants en cas de chute. Il est malheureusement fréquent que la victime oublie ou n'ait pas conscience qu'elle porte un tel dispositif. Elle ne pense donc pas à alerter les aidants par ce biais, ce qui en constitue le principal désavantage.

D'autre part, des solutions intégrées de télé-vigilance directement dans l'habitat sont aussi disponibles (e.g. EDAO⁷ [82]). Cependant, ces technologies ont tendance

3. *Plumbing to Prevent Domestic Hot Water Scalds* (Article en ligne) - <http://www.hgexperts.com/article.asp?id=5135>

4. Robi-Light - <http://www.gerontechnologie.net/indicateur-de-temperature-deau-au-sortir-du-robinet>

5. High Tech Place - <http://www.hightechplace.com/electronique/eclairage-led/ampoules-led/embout-robinet-lumineux-led-3-couleurs.html>

6. Luisina - <http://www.luisina.com/>

7. Edao - <http://www.edao.com/domicile/edao-system/>

1.2. GESTION DES RISQUES À DOMICILE

à utiliser de la surveillance vidéo, ce qui tend à freiner l'acceptation du dispositif de la part de l'occupant.

Intoxication par les gaz et fumées Aucune technologie n'a été identifiée au cours de cette revue de littérature pour prévenir les intoxications. Néanmoins, certains dispositifs initialement destinés à monitorer d'autres risques peuvent remplir partiellement cette fonction.

De cette manière, les détecteurs de fumée permettent d'éviter l'intoxication en cas d'incendie [5], alors que les capteurs physiologiques de certains dispositifs de télé-vigilance peuvent informer de l'inactivité d'une personne en cas d'intoxication conduisant jusqu'à l'évanouissement du sujet.

Notons aussi que l'utilisation de dispositifs d'extraction tels que les hottes de cuisine permettent de limiter les émanations et ainsi réduire les risques d'intoxication pour une personne vulnérable.

Suivi de la personne

Bharucha *et al.* [10] proposent une revue de la littérature des produits d'assistance existants ainsi que des projets en cours de développement et destinés spécifiquement aux personnes atteintes de démences. On retrouve alors nombre de capteurs environnementaux, bio-senseurs, orthèses cognitives, capteurs physiologiques, ou encore capteurs intégrés dans nos artefacts de tous les jours. Pour chacun des éléments recensés, la population ciblée, le niveau de maturité du produit, l'inclusion (ou non) de personnes démentes dans le processus de développement, ainsi qu'une critique, sont proposés.

Lou *et al.* [58] proposent eux aussi un état de l'art des dispositifs électroniques existants en les regroupant en quatre grandes catégories : les aides-mémoire, les assistants à la réalisation d'AVQ, les aides à la sociabilisation, et enfin les dispositifs pour la sûreté de la personne.

Cependant, ces dispositifs ne sont pas spécifiquement destinés à améliorer la sécurité à domicile, mais répondent plus à une problématique de suivi du patient (e.g. médication, données sur la santé) afin d'alléger la charge des aidants. Ils permettent ainsi de traiter les risques additionnels présentés en 1.1.3.

Habitats intelligents

Le terme *habitat intelligent* est utilisé pour désigner un habitat équipé de capteurs et d'actuateurs afin de surveiller les activités de ses occupants, et interagir avec eux en leur proposant des services adaptés et ainsi les soutenir dans leur démarche de maintien à domicile [41].

Avec l'utilisation de l'intelligence ambiante et de l'informatique diffuse, on ouvre alors de nouvelles perspectives en terme d'assistance à domicile. De ce fait, la communauté scientifique s'intéresse tout particulièrement à ce domaine, et les recherches qui s'y rapportent sont en pleine expansion.

Pour appuyer ce point, Demiris *et al.* [24] proposent une revue des principaux laboratoires de recherches oeuvrant dans ce secteur et disposant d'un prototype fonctionnel d'habitat. Pour chacun d'eux, ils précisent la population ciblée ainsi que les principaux travaux qui y sont associés (aussi bien achevés qu'en développement).

Chan *et al.* [16] complètent ce travail de synthèse, et proposent une revue particulièrement bien détaillée des défis actuels et futurs à relever dans ce secteur. Malgré la très large couverture de cette étude, il s'avère que les accidents domestiques ne sont que très peu adressés. Seules les chutes sont brièvement mentionnées.

1.2.4 Technologies spécifiques à l'utilisation de la cuisinière

Les technologies les plus communes pour limiter les incendies et les brûlures dans l'habitat ont été présentées en 1.2.3. Malgré les limites qu'ils présentent en terme de maintien et fiabilité à détecter les départs de feu réels [51, 5], les détecteurs de fumées s'imposent actuellement comme étant la technologie dominante.

Cependant, avec l'intégration massive de l'informatique dans nos artefacts de la vie courante, les cuisinières intègrent maintenant régulièrement des dispositifs d'arrêts automatiques après une durée prédéfinie [60], ou encore permettant de contrôler à distance l'appareil électroménager⁸ [89]. Ces fonctionnalités ont plus une finalité pratique qu'un but sécuritaire, mais peuvent toutefois contribuer à éviter des accidents. Néanmoins, les cuisinières les moins récentes ne disposent pas de ces fonctions, c'est pourquoi il existe des produits les proposant séparément.

8. Pour exemple : Fours intelligents TMIO - <http://www.tmio.com/products/>

1.2. GESTION DES RISQUES À DOMICILE

D'autres réactions de prévention des accidents peuvent être déclenchées à l'aide de dispositifs satellites eux aussi disponibles dans le commerce.

La liste ci-dessous illustre les types de produits proposés :

- La gamme *SafeCook*⁹ [50], qui permettent non seulement de limiter le temps d'utilisation de l'élément de cuisson, mais aussi d'en arrêter automatiquement l'utilisation après un temps prédéfini. L'arrêt est effectué en coupant l'alimentation de la cuisinière (électricité ou gaz). Un autre exemple d'un tel système : la *HomeSenser* de la compagnie *HomeSense*¹⁰ [31].
- Les produits *Stove Top FireStop*¹¹ [32] détectent automatiquement les départs de feu sur les ronds de cuisson. Ils se présentent sous la forme de dispositifs électroniques à placer au niveau de la hotte de cuisson, et intégrant une capsule d'agent extincteur qui est vaporisé sur la cuisinière dès la détection de l'incendie.
- Le *Stove Guard*¹² [59] combine le dispositif d'arrêt de la cuisinière avec un second boîtier comprenant un détecteur de mouvements et un *timer* programmable. Ainsi, dès que la durée d'inactivité à proximité de la cuisinière excède celle qui a été programmée, la cuisinière est arrêtée automatiquement.

Parmi les éléments énoncés ci-dessus, nous remarquerons que seuls les incendies sont traités. Cependant, les brûlures tendent à diminuer lorsque les usagers utilisent des plaques chauffantes se basant sur le principe d'induction. Cela s'explique par le fait que ces dispositifs ne chauffent que lorsqu'un récipient adéquat est présent dessus. On évite donc ainsi les brûlures sur des ronds libres laissés en fonctionnement.

D'autre part, seuls deux projets de recherche concernant la sécurité lors de l'utilisation de la cuisinière ont pu être identifiés :

- La *U.S. Fire Administration* mène depuis 2003 un projet¹³ [4] destiné à développer des capteurs capables de détecter les variations de température de la

9. <http://www.safecook.ca/>

10. <http://www.homesensers.com/>

11. <http://www.stovetopfirestop.com/>

12. <http://stoveguard.ca/>

13. http://www.usfa.dhs.gov/fireservice/research/other/other_stove.shtm

plaque céramique de la cuisinière (pour les appareils qui en sont pourvus). Ces capteurs sont situés directement entre l'élément chauffant et la vitre céramique. De cette manière, lorsqu'une substance se répand sur la surface lors de l'utilisation d'un rond chauffant, l'élément est arrêté automatiquement afin de prévenir un éventuel départ d'incendie.

- En 2011, Wai *et al.* [90] proposent un projet de système intelligent destiné à assurer la sécurité des personnes âgées seules, et ce au sein de la cuisine de l'habitat. En se basant sur l'informatique diffuse, ils proposent ainsi d'y installer un réseau de capteurs et d'effecteurs, pour être en mesure par la suite de détecter les situations anormales lors de la réalisation des tâches les plus courantes (e.g. vaisselle, préparation de repas, prise de médicaments), et fournir une réaction appropriée.

1.3 Positionnement du projet Inovus

Cette section synthétise l'état de l'art sur les risques à domicile présentés en 1.1 et les moyens de gestions associés, présentés en 1.2. L'objectif ainsi visé est de faire ressortir les lacunes en terme de solutions technologiques de prise en charge. Ce bilan nous permettra ensuite de justifier de la portée du projet INOVUS.

1.3.1 Synthèse sur les risques à domicile

Cette revue de littérature a permis d'identifier les principaux risques présents au domicile de la personne âgée ou démente. Cette revue a de plus permis d'étudier précisément les moyens mis en oeuvre pour lutter contre ces risques. Il en est ressorti deux volets essentiels, ayant une finalité différente l'une de l'autre :

- On notera donc d'une part l'aspect prévention, par le biais des outils d'évaluation des risques directement au domicile. Cette prévention est renforcée par les différentes campagnes de sensibilisation menées par les principaux acteurs du maintien à domicile.
- D'autre part, les technologies d'assistance destinées à contribuer au maintien à domicile : elles sont certes en pleine expansion. Cependant, elles occupent

1.3. POSITIONNEMENT DU PROJET INOVUS

très souvent le rôle d'orthèses cognitives et ne considèrent pas suffisamment les dangers présents au domicile.

Le tableau 1.1 présente la synthèse des risques identifiés dans cette revue de littérature, de leurs contextes d'apparition majeurs, et des solutions technologiques existantes.

Les nombreux travaux étudiés lors de cette revue ont fait ressortir la cuisine comme étant l'une des pièces les plus à risques dans le domicile. Pour rappel, Ehrlich *et al.* en 2005 [30] indiquent que, sur l'ensemble des cas de brûlures recensés lors de leur étude, 68% sont survenus lors d'activités de cuisine, et Hall [43] présente les activités de cuisine comme étant la première cause d'incendies domestiques.

Or, nous pouvons remarquer dans ce tableau qu'en ce qui concerne les quatre risques majeurs présentés en 1.1; l'utilisation de la cuisinière représente le seul contexte d'apparition commun pour trois d'entre eux (incendies, brûlures, intoxication). De ce fait, la cuisinière constitue donc un élément clé en terme de sécurité à domicile, et ce point se trouve renforcé dans le cas de personnes présentant des troubles cognitifs.

Cet état de l'art a de plus montré que les brûlures et les incendies sont les accidents les plus à même de survenir lors de la réalisation d'activités utilisant la cuisinière. Ces dangers sont rejoints par l'intoxication, qui peut survenir dans certains contextes, et constituent donc à eux trois le panel de risques nécessitant une intervention.

Nous remarquerons enfin dans le tableau 1.1 que les technologies associées à chacun de ces risques présentent certaines limitations qui restreignent leur efficacité :

- les informations fournies à l'utilisateur sur la présence d'un danger sont quasi nulles, et pas toujours adaptées aux déficits de la personne ;
- la réaction offerte, lorsqu'elle existe, se contente généralement soit d'une alarme sonore, soit d'un arrêt général de la cuisinière ;
- lorsqu'une détection d'activité est effectuée, elle se concentre uniquement sur l'environnement direct de la cuisinière, et non pas sur l'ensemble de l'habitat ;
- enfin, chacune des technologies actuellement offertes ne couvre qu'un seul risque.

CHAPITRE 1. REVUE DE LA LITTÉRATURE

Compte-tenu des lacunes identifiées tout au long de cette étude, le projet INOVUS a pour objectif de fournir une solution de plus large couverture en terme de gestion des risques liés à l'utilisation de la cuisinière, et ce en passant par le biais des technologies de l'habitat intelligent.

Tableau 1.1 – Synthèse des risques à domicile et des processus de gestion associés, tels que traités dans la littérature.

Risques	Contextes	Prise en charge	Références
Brûlures	Toilette Utilisation cuisinière Four micro-ondes Vaisselle, Cigarette	Limiteurs de température Indicateurs lumineux (surfaces chaudes)	[47, 30, 55]
Incendies	Utilisation cuisinière Cigarette Dysfonction électrique	Détecteurs fumées/gaz, extincteurs auto-déclenchés, timers et dispositifs d'arrêt auto. pour cuisinière	[51, 55, 65] [91, 13, 43] [5, 29]
Intoxication (gaz/fumées)	Utilisation cuisinière Propagation incontrôlée	Dispositifs d'extraction (tels qu'une hotte d'aspiration) Détecteurs de fumées ou de gaz	[9, 13, 55] [78]
Chutes	Déplacements Escaliers, Mobiliers Sols mouillés Éclairage inadapté	Détecteurs de chutes, télévigilance, surveillance vidéo, éclairages spécifiques	[13, 62, 21] [39, 22, 88] [54, 70]
Médication	Erreurs de dosage Erreurs de médicament Oublis	Piluliers électroniques et autres dispositifs de rappels	[13]
Alimentation	Tâches de cuisine	Monitoring des aliments consommés	[18, 20, 92]
Errance et désorientation	Déplacements ou activités complexes	Assistants de guidage, dispositifs de localisation, guides lumineux, balises RFID, détecteurs mouvements/ouverture de porte	[81, 88]

1.3.2 Apports du projet Inovus

Nous avons précédemment montré que la problématique de la sécurité de la personne démente à son domicile est primordiale, en particulier lors de la réalisation de tâches de cuisine utilisant la cuisinière.

Il a aussi été montré que les solutions technologiques proposées actuellement peinent à répondre à l'ensemble des besoins qui y sont liés.

L'utilisation des infrastructures de la maison intelligente ouvre de nouvelles perspectives en terme de sécurité à domicile.

En effet, le déploiement des capteurs et actuateurs dans l'ensemble de l'habitat permet d'effectuer un suivi temporel et spatial de la personne.

De plus, la variété de capteurs déployables dans l'environnement permet une détection plus fine des différents contextes de risques. L'utilisation de l'intelligence ambiante pour traiter les données enregistrées permet de monitorer l'environnement de l'utilisateur, et ce d'une manière qui lui est totalement transparente : il n'a aucun dispositif additionnel à gérer, et il se contente simplement d'utiliser les ressources de la cuisine comme il le fait à son habitude.

Par nature, un habitat intelligent intègre une grande variété d'actionneurs, et l'interaction homme-machine s'en trouve alors grandement améliorée : les moyens d'informations avec l'utilisateur peuvent être aisément adaptés aux déficits cognitifs, là où les dispositifs actuels fournissent parfois des informations non perceptibles pour l'utilisateur déficient.

Enfin, en intégrant les avancées actuelles de reconnaissance d'activités et d'intelligence artificielle, l'éventail des réactions possibles en cas d'apparition de risques se trouve grandement élargi : plus qu'un simple arrêt de la cuisinière, il devient alors possible d'effectuer un raisonnement sur les actions passées, courantes, et futures de l'utilisateur, et ainsi prendre des mesures plus adaptées. Nous pourrions par exemple imaginer simplement réduire la puissance de l'élément de cuisson en utilisation, ou encore démarrer automatiquement la hotte d'aspiration à partir d'un certain seuil de dégagement de fumée.

Il devient aussi possible de décider automatiquement de contacter les aidants ou services d'urgence en cas de situations critiques, sans passer par un service de télévi-

1.4. USAGE DE LA SPÉCIFICATION FORMELLE

gilance.

En se basant sur ces constatations, on remarque que les possibilités offertes par les habitats intelligents permettent d'atteindre une couverture de risques jusque là inégalée en terme d'étendue des contextes, de modularité, de traitement, ou encore de réactions faces aux risques.

Le projet INOVUS a donc pour but d'explorer ces possibilités dans le cadre des activités liées à l'utilisation de la cuisinière, qui, comme démontré précédemment, constitue l'une des sources majeures d'accidents au domicile des populations ciblées. Les travaux présentés dans ce mémoire constituent donc les premières étapes actuellement réalisées dans le projet.

1.4 Usage de la spécification formelle

Afin de valider le logiciel produit, les différents processus de développement informatique intègrent différentes phases de tests logiciels. Ces phases se placent aux différents niveaux du développement, des tests unitaires aux tests de validation (tests systèmes), en passant éventuellement par de la simulation. Cela permet de parvenir à une couverture de tests généralement satisfaisante pour le produit final en assurant son bon fonctionnement.

Cependant, comme le précisera le chapitre 2, malgré l'apport indéniable et incontournable de ces procédures, il apparaît fréquent que des erreurs subsistent dans le développement logiciel, ce qui peut parfois entraîner des conséquences graves. Ce point est d'autant plus marqué lorsqu'il s'agit de systèmes critiques tels que le projet INOVUS. Les méthodes formelles de spécifications semblent alors toutes indiquées pour venir pallier à ces manques.

De manière générale, les méthodes formelles permettent d'apporter un raisonnement logique sur un système, afin d'en vérifier certains éléments de manière mathématique. Elles se déclinent en différents outils parmi lesquels on trouve notamment la vérification de modèle (*model checking*).

Le modèle de solution développé dans ces travaux de maîtrise ayant fait appel à cette

technique afin de vérifier la détection de risques et l'interaction avec l'utilisateur, cette section présente les principales utilisations qui ont pu être recensées dans les domaines connexes, ainsi que l'utilisation qui en a été faite pour la couche logicielle du projet INOVUS.

1.4.1 Utilisations dans des domaines connexes à la couche logicielle du projet Inovus

La littérature actuelle est très restreinte quant à l'utilisation des méthodes formelles en informatique diffuse, néanmoins cette revue a permis d'en identifier les quelques usages qui en ont été fait dans les domaines des services de santé et/ou de l'informatique diffuse au cours de dernières années. Les principales approches représentatives sont décrites ci-dessous.

Hoareau [48] présente un système de localisation de la personne dans un espace intelligent. Pour améliorer le traitement des requêtes de positionnement, il intègre un raisonnement basé sur du model checking.

Herzberg *et al.* [46] ont développé une application logicielle de suivi à distance entre patients et médecins. Les méthodes formelles ont alors été utilisées pour la phase de spécification de leur système.

Enfin, Chen [19] propose quant à lui une utilisation du model checking spécifique aux habitats intelligents utilisant le cadre de gestion de composants *OSGi*. En effet, il suggère l'utilisation de cette technique pour gérer les changements de références entre les services lors du cycle de vie des différents composants d'une application. Selon l'auteur, cela permet d'éviter la perte inopinée d'un service, et ainsi laisser à disposition de l'utilisateur une ressource qui peut lui être potentiellement vitale.

1.4.2 Utilisation pour la couche logicielle du projet Inovus

La problématique de sécurité à domicile traitée par le projet INOVUS engendre des composantes de fiabilité très importantes : le système doit répondre impérativement

1.4. USAGE DE LA SPÉCIFICATION FORMELLE

aux exigences de détection des contextes dangereux et de génération de réactions adaptées. En effet, un danger doit être traité dès son apparition, et la surveillance, l'information, ainsi que la réaction qui en découlent doivent être automatiquement initiés. Il est aussi nécessaire de s'assurer que les informations et réactions générés sont bien adaptées au contexte, l'objectif étant d'assurer de manière effective la sécurité de la personne et donc de ne pas lui fournir d'information erronée.

Afin de satisfaire ces exigences, il a été décidé d'utiliser la vérification de modèle. Cet outil nous permet ainsi de s'assurer que le modèle de solution élaboré satisfait pleinement les exigences critiques en terme de détection et de réaction, et ce en imposant la cohérence du système ainsi que la vérification de propriétés (exigences critiques).

Les travaux présentés dans les sections précédentes de ce chapitre ont permis d'identifier les risques majeurs à traiter lors de l'utilisation de la cuisinière. Les contextes d'apparition de ces dangers sont nombreux au quotidien et nous avons montré précédemment que les populations ciblées par le laboratoire Domus sont particulièrement sujettes à ces dangers. Dans le cadre de ces travaux de maîtrise, seul un échantillon de contextes à risques a été choisi, en se basant sur les situations les plus susceptibles d'apparaître lors de la réalisation d'AVQ.

Pour chacun de ces contextes, il a été nécessaire de définir un algorithme de détection, ainsi qu'une stratégie de réaction. La spécification formelle a donc été utilisée afin de s'assurer de la fiabilité de ces algorithmes et stratégies.

Les chapitres 2 et 3 présentent plus en détails cette utilisation.

CHAPITRE 1. REVUE DE LA LITTÉRATURE

Chapitre 2

Approche de conception logicielle pour le projet Inovus

Ce chapitre présente la première publication concernant le projet INOVUS. Cet article a été accepté pour le workshop **SmartE** de la conférence **PerCom 2011** (*9th IEEE International Conference on Pervasive Computing and Communications*, 21-25 Mars 2011, Seattle, USA).

Cette publication décrit le modèle de solution proposé dans le cadre du projet INOVUS. Ce modèle a été renforcé par l'utilisation du vérificateur de modèle ALLOY afin d'en assurer sa cohérence, ainsi que le respect des exigences à satisfaire. L'utilisation de spécifications formelles permet de plus de vérifier le bon respect des spécifications essentielles du projet. Cet article présente donc l'insertion de la vérification de modèle (*model checking*) avec ALLOY dans le processus de développement d'une application de sécurité de la personne dans une cuisine intelligente. Le modèle décrit ici correspond à la première version de l'application qui est souhaitée dans le projet INOVUS.

CHAPITRE 2. APPROCHE DE CONCEPTION LOGICIELLE

Contribution de l'étudiant

Cet article a été rédigé conjointement par *Thibault de Champs* et *Mohammed Ouenzar* dans le cadre de leurs travaux de maîtrise. Les éléments présentés ainsi que leurs corrections éventuelles ont été appuyés par *Bessam Abdulrazak* et *Marc Frappier*, directeurs de recherche des deux étudiants, ainsi que par *Benoît Fraikin* (co-directeur de *Mohammed Ouenzar*). Leur contribution s'étend de plus à la révision des différentes versions de l'article, et est accompagnée par celle d'*Hélène Pigot*, co-directrice de recherche de *Thibault de Champs*.

Versions antérieures

Non applicable.

Pervasive Safety Application with Model Checking in Smart Houses: the Inovus Intelligent Oven

Thibault de Champs¹, Mohammed Ouenzar²
Bessam Abdulrazak¹, Marc Frappier²
Hélène Pigot¹, Benoît Fraikin²

¹ Laboratoire DOMUS, ² Laboratoire GRIL
Département d'informatique, Université de Sherbrooke
Sherbrooke, Québec, Canada J1K 2R1

Keywords: smart house; pervasive computing; home safety; assistive technology; model-checking; alloy;

Abstract

Safety is a major challenge in developing assistive software for people with special needs in smart houses. INOVUS is an ongoing project about safety issues of cooking activities. This paper presents the INOVUS project and highlights lacks in current software development processes to meet safety requirements in pervasive computing. In the INOVUS project, we propose an approach to introduce model checking using ALLOY as a new layer to strengthen the design and the understanding of specifications in the development process of safety related applications in smart houses. Finally, we illustrate the model checking process with ALLOY and present the initial results of the INOVUS prototype.

CHAPITRE 2. APPROCHE DE CONCEPTION LOGICIELLE

2.1. INTRODUCTION

2.1 Introduction

Today, pervasive computing has become an active research area in which building smart houses to fulfill basic needs of *Activities of Daily Living* (ADL) is emerging. In smart environments, applications for cognitive assistance aims to provide *People with Special Needs* (PwSN¹) with adapted help. Safety is a major challenge in developing assistive applications for PwSN in smart houses. This is an important aspect of the INOVUS project, which aims at developing a safety solution for people when performing cooking activities in a smart house. The INOVUS pervasive application ensures PwSN safety based on defined risk contexts, offers adequate actions in case of upcoming or inevitable hazard and provides information about potential current risks to the inhabitant and the caregivers if it is needed.

Human environment interaction is considered as one of the predominant part in pervasive computing. Promoting independent living for PwSN in smart houses requires ensuring safe human-machine interaction and safety in general at home, and consequently reliable systems. To achieve this goal, usual software development processes [74, 75] are incomplete in term of software consistency management [71]. This paper proposes an additional layer based on formal specifications to ensure consistency of the software under development, which is a key point for its future reliability. Formal specifications are mathematical software representations, which describe system objectives and behavior early in the development. Additionally, these methods allow one to verify/prove the resulting model [8, 3]. In a pervasive computing approach, model checking stands out because of its ease of use and automation.

This paper discusses the INOVUS design, introduces our approach using model checking based on first-order logic and presents the implementation of the solution using ALLOY [52] to develop a software capable of managing the potential risks in a smart kitchen. The goal is to improve the design phase in order to facilitate development of applications, in addition to strengthen the consistency in the development process of safety application. This paper also presents the limits of our approach and alternative improvements.

This paper is structured as follows: Section 2.2 introduces the INOVUS project

1. People with Special Needs: Elderly or people with disabilities.

and raised issues. Section 2.3 presents the related work on consistency checking. Section 2.4 discusses the proposed additional layer based on model checking and the INOVUS implementation using the ALLOY model checker. Section 2.5 discusses the results and Section 2.6 concludes on the proposed approach.

2.2 Inovus Intelligent Oven Project

This section presents the INOVUS project: the context of the project, an illustrative scenario, an overview of the specifications and the adopted approach.

2.2.1 Context

INOVUS is a specific safety-related ongoing project in DOMUS laboratory² [25]. The DOMUS lab aims at research in domotics, mobile computer science, and specializes in the development of assistive technologies for PwSN within a smart environment. These environments provide PwSN with assistance in terms of services, medical monitoring, safety, among others.

Safety at home is an emerging issue in pervasive computing. It is related to various ADLs. Studies have shown that this is a major concern for PwSN and their caregivers [40, 86]. These risks are mainly located in the bathroom and the kitchen [15, 62, 84]. For instance, main risks at home for PwSN are falls, fires and burns [15, 44, 47, 62, 84]. Numerous studies/solutions have addressed the fall problem [10, 54, 68]. However, rare are those addressing fires and burns. The existing solutions address risks after manifestation or propose prevention guides (e.g. [5]). Thus, there is no integrated solution taking into account the major contexts of risks for cooking activities. This justifies the motivation behind the INOVUS project.

2.2.2 Scenario

In order to clarify how to address the aforementioned issues, we examine the following scenario. Mr. Smith is a 74-year-old man who lives alone. One year ago,

2. <http://domus.usherbrooke.ca/>

2.2. INOVUS INTELLIGENT OVEN PROJECT

he was diagnosed with probable Alzheimer disease in the early stage. To facilitate his daily living, his daughter (a caregiver) visits him everyday at noon to prepare lunch and dinner meals for him. Two month ago, as his daughter was late, Mr. Smith started preparing food using the oven. Few minutes later, he moved to the living room to watch his favorite TV show which just began, forgetting the meal on the stove. His daughter arrived forty minutes later. As she entered the house, she noticed a dense smoke because of the food burning and her father hardly breathing.

To help Mr. Smith and his daughter, we propose an integrated technological support to ensure safety in cooking activities. In the above case, the INOVUS intelligent oven would have pro-actively replied to the situation. (1) The too long time away from the oven would have been detected. A warning would have been sent to Mr. Smith via TV, screens, and/or speakers of the smart environment. If Mr. Smith would have not reacted in time to this message, the system would have automatically shut down the oven. (2) In the same time, the smoke appearance would have been detected. This critical risk which threatened Mr. Smith would have resulted in an emergency stop of the oven. In addition, his daughter or/and relevant emergency services (e.g. 911) would have been informed about the situation. Such a solution would ensure safety of Mr. Smith and serenity for his daughter.

2.2.3 Specifications

The goal of the INOVUS project is to provide an autonomous solution that ensures user safety and takes in consideration major safety issues in cooking activities. INOVUS (architecture in Figure 2.1) is based on smart environment infrastructure, especially sensors and actuators networks distributed in the kitchen area. Sensors allow the system to infer the activities of the house (for instance movements near the oven or actions on it) or detect changes in the surrounding environment (e.g. smoke, temperature of surfaces, people presence). Actuators allow the system to provide feedback to users through screens, speakers, or flashing lights, or to control appliances (shutdown the oven).

The INOVUS *decision engine* observes sensors in order to infer risk contexts, informs the user about existing hazards through actuators in the house (warnings) and

alerts caregivers if it is necessary (emergency call). The user has a pre-configured time to react to the danger or inform the *decision engine* that he/she is conscious of the situation. If the user does or can not react to the warning, the *decision engine* automatically takes control of the oven and plans actions in order to avoid dangerous situation. In case of an unavoidable accident, emergency procedures are initiated.

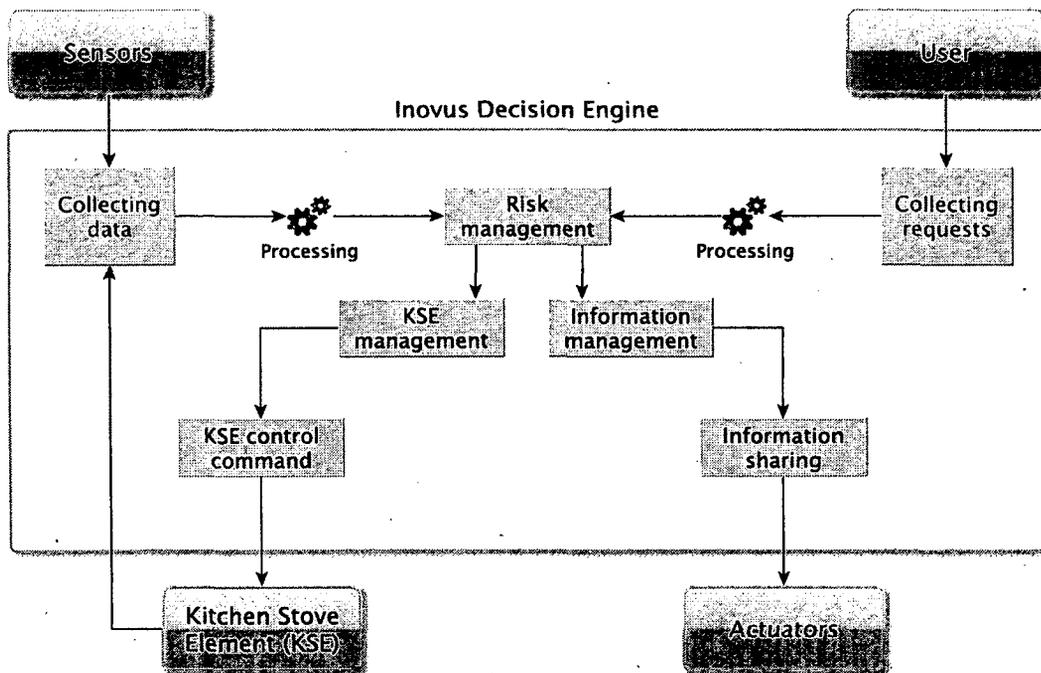


Figure 2.1: The INOVUS architecture in the home environment.

2.2:4 Approach

A software development process includes several steps with specific objectives for the development of an application. The typical main steps are requirements collecting, specifications writing, design, implementation and tests [74, 75]. Reliability of the resulting software is evaluated according to verification and validation steps. Despite such a relatively complete process, errors or system failures often remain and occur throughout software's life cycle. Software consistency and reliability must be ensured when addressing safety of people in ADLs at home. Evolving directly in the

2.3. RELATED WORKS

occupant's environment and assisting him/her in performing ADLs, the application should not provide erroneous information or generate incorrect reaction that can lead user to dangerous situations. In order to reduce the risk of introducing faults in software, the INOVUS software development process has been strengthened with formal specifications. To do so, we added a model checking phase into the development process. This additional step consists in developing a formal specification (i.e. model) of the system, which is then validated to ensure its adequacy with system requirements and verified to check formal properties. The formal model is developed in conjunction with the classical software design phase, creating a loop between these two steps (Figure 2.2). This cycle is located after the writing of specification and before the implementation (coding), which allows better validation of the design. This combination of standard software design and model checking adds a consistency layer to the resulting model. The model checking approach provides a mathematical verification of the model. The efficiency of this approach has been proven [8], and coupling it with one of the usual tools for software design ensures that the requirements of the project are satisfied in terms of software behavior. This appears to be a key point while designing applications for smart houses.

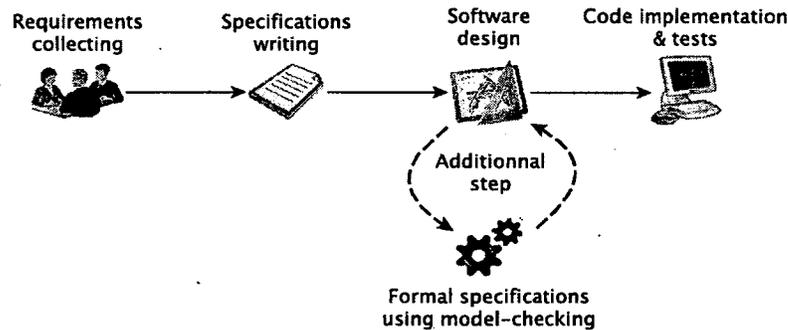


Figure 2.2: Classical process with the proposed additional step.

2.3 Related Works

This section highlights more details about formal specifications and consistency checking in the software development process.

2.3.1 Consistency Issue in Software Development Process

There exist several software development processes (e.g. waterfall, iterative, spiral among others) [42, 74, 75]. The phases of these processes are quite similar; design, testing, debugging and verification. Software design appears to be an important phase of the process. Experts view design consistency as a lack in the management of software development process [34, 42, 45, 49, 71]. Huzar et al. in [49] defines consistency as incorrect relations between the artifacts that represent the aspects of a system, and differentiates intra-model and inter-model consistency. On the other hand, Nuseibeh et al. in [71] defines inconsistency as "*any situation in which a set of descriptions does not obey some relationship that should hold between them.*" Most description notations, especially the *Unified Modeling Language* (UML), use multiple views to represent the software architecture according to the specifications with imprecise semantics, and thus suffer inconsistency. Nuseibeh et al. in [71] defends that maintaining consistency all the time is counterproductive, and time and resource-consuming. They proposed a framework to monitor, diagnose, handle, measure and analyze inconsistent elements in order to obtain respectable and manageable inconsistency.

Moreover, several description techniques were proposed [28, 34, 45, 49] where formal approach remain a recurrent suggestion because of its mathematical basis. Formal specifications provide precise models, support the specification of constraints and check them. Model checking is the formal approach adopted in the INOVUS project. As explained in the next section, the best way to manage inconsistencies with formal specifications lies in finding appropriate abstractions.

2.3.2 Formal Specifications

Formal specifications foster abstraction as the core of software development [52]. It is recurrent that developers come up with improper abstractions that result in inconsistent design and laborious implementation. It is the result of a lack of abstraction and modeling in common software development process. Modeling is often based on *wishful thinking*, a concept taken up by B. Meyer [67] who describes the formation of beliefs and decisions made according to what might be pleasing to imagine

2.3. RELATED WORKS

instead of by appealing to evidence, rationality or reality. Formal specification has been proposed as a solution to tackle the lack of abstraction by addressing it early in the process. Some examples [8] show that the use of formal specifications often comes up in very high critical situations and later in the software life cycle, and corroborate its efficiency. The under-usage of formal specifications until now is explained by two facts: complex mathematical logic notation and slow theorem provers development. Model checking [8] is one of the main dimensions in formal specification. Model checkers explore the transition system graph to check the satisfaction of desired properties. Numerous model checkers have been developed by organizations for specific purposes, and the difficulty is to find the most appropriate one depending on the needs. Frappier et al. in [35] presents a comparison of common model checkers to address this difficulty and find out about the benefits and disadvantages of each model checker.

ALLOY is the model checker used in the INOVUS development process. ALLOY is a model checker based on first-order logic which offers a modeling language close to *Object-Oriented Languages* (OOL). ALLOY's salient features are based on mathematical techniques (e.g. solvers) to check model consistency and find counterexamples of assertions with an automatic analysis. ALLOY also provides tools to evaluate expressions and formulas on models, or give a graphical representation of the model under development.

In software testing, the number of possible test cases can be huge, thus unmanageable. Consequently, tests are conducted only on the cases considered by the software tester as the limit points, which means that there are numerous interesting cases that are not tested, which can be a source of defects. In ALLOY, the space of checked cases is significantly larger than the test space that is typically covered by manual testing because of the results of research in constraint-solving technology [64]. ALLOY analysis is not complete since it only examines a bounded space according to the scope selected by the user. On the other hand, manual testing can address specific scenarios which are outside the state space that can be fully analyzed by ALLOY, due to memory constraints. Thus, the approaches are complementary.

2.4 Inovus Development Process

Section 2.2 described the INOVUS context and requirements. Following related work mentioned in previous sections, we describe here the first prototype of INOVUS and the corresponding implementation using ALLOY to ensure consistency. A typical approach in formal specifications is incremental development. This allows to begin the software design at a high abstraction level in order to validate the main concepts of the desired model. Subsequent increments refine the software design, gradually reducing abstraction levels. Such a technique let designers validate step by step the software under construction. The INOVUS design is based on this method.

This section describes the first increment in terms of specifications, design, and implementation with ALLOY.

2.4.1 First Increment Specifications and Design

The first increment specifications are based on the requirements described in Section 2.2. This first version includes a small subset of the risks, and it will be extended in subsequent increments with a larger cooking hazards collection. Table 2.1 describes the integrated risks and their different contexts of materialization.

Table 2.1: Risks considered in the first increment.

Risks	Contexts
Burns	Hot surfaces (oven front or hotplate)
	Opened oven door
Fires	Food forgotten while baking
	No surveillance in home
	People leaving home while food baking
Intoxication	Smoke detection in the kitchen

This preliminary prototype follows the architecture described in Figure 2.1. We handle the crucial design step with UML due to its object-oriented approach and its multiple software views. Static diagrams can be easily represented in the structure

2.4. INOVUS DEVELOPMENT PROCESS

of the ALLOY model and dynamic diagrams concur with dynamic ALLOY features. Consequently, we have to highlight the entities composing the system and the events (preconditions, actions, and post-conditions) that may occur. To do so, relevant diagrams for this step are selected and class, use case, and state diagrams are created.

Stove elements (i.e. *hotplates* and *oven*) are distinguished, and commonly denoted as *Kitchen Stove Element* (KSE). *Infrared sensors* are used to localize the user. *Smoke sensor* prevent fires. *Contact* and *temperature sensors* represent the KSE state. Moreover, *timers* measure use time or no surveillance duration for a KSE, or time with nobody at home. Sensors data is processed by the *decision engine*, which provides adequate *information* through *actuators* (*screen*, *light* or *speaker*), and reacts accordingly to the detected *risk situation* and user's requests.

2.4.2 Implementation and Analysis Using Alloy

Following the UML design, the ALLOY implementation is facilitated. The design brings out static and dynamic parts of the software. Static diagrams resulting from the design stage are easily translated into ALLOY code. UML dynamic models lead to the definition of possible events in the ALLOY model. These events represent user actions, environmental changes, or decision engine reactions.

The resulting ALLOY model is analyzed by the ALLOY analyzer, which verifies the model consistency (in terms of first-order logic) and gives *traces*. A *trace* is a sequence of successive states generated by ALLOY according to a defined scope. These states result from the evaluation of transition system predicates.

Some abstractions had to be done. In particular, the way to interact with the user of the kitchen stove has been generalized and the message that the system is sharing with the user is not specified (i.e. only actuators states are checked). The two other needed abstractions concern time and temperature evolution. This simplify the formal design and does not affect the consistency of the resulting model instance.

This constitutes the first step of model validation, which ensures the model consistency, by looking at tracing produced by ALLOY. The second step is to check properties on the model. Properties checking is a step provided by model checkers to verify that the specification entails the desired properties.

Properties must be first clearly described in natural language. Then they are translated into first-order formulas and checked with ALLOY. ALLOY verifies that the properties hold for all (bounded) models of the specification. For instance, this amounts to checking that all finites traces of length k (i.e. the bound selected by the user) satisfy the required property.

Concerning relevant properties selection, they are classified according to the severity/degree of the hazard they represent. The INOVUS project focuses on safety for people while cooking, therefore we want to validate the safety properties that the model has to satisfy. For this reason, we need to choose the ones which appear the most relevant to safety issues. Concretely, we want to especially validate detection of risk situations, and corresponding reaction. These assertions represent the system's behavior when a risk situation occurs.

This section presented the two main components of a formal model. The formal model is analyzed by ALLOY in order to ensure its consistency. Critical specifications fulfillment is ensured by properties checking. Formal representation of the system gives a new abstraction layer that brings out some inaccuracies in the structural design or in specifications. An implementation based on the UML design strengthened with the ALLOY model will ensure that safety properties will be verified by the resulting consistent software.

2.5 Results and Discussion

The development of the first INOVUS prototype allowed us to evaluate the use of ALLOY, and as a consequence model checking, for the development of a safety application in a smart environment. In this section, we highlight the main results of our evaluation, and we hope that they can be relevant for pervasive computing applications in general.

2.5.1 Relevancy of the Proposed Process

Formal model helps understanding the specifications for a better code implementation. The proposed process is iterative (Figure 2.2). Each cycle of the formal model

2.5. RESULTS AND DISCUSSION

design highlights mistakes or improper elements of the UML design according to the specifications. This enables adjustments in the initial UML representation and/or the formal model. Nine iterations were necessary to produce a complete and stable instance of the model first increment in INOVUS project. Each iteration results from poorly implemented specifications in the structural model, or from new abstractions which appears as necessary in the formal model.

Choice of elements that can be omitted is fundamental for a good abstraction. There is no need to represent the system in details but only the elements which need to be checked with ALLOY, which has two main advantages. First, the model checker workload is substantially reduced whereas a too large model would be time consuming and resource intensive. The model checker may be unable to check too large specifications. Second, by verifying step-by-step the critical parts of the design with the model checker, the corresponding code implementation will be facilitated. In fact, a complete representation of the specifications in the formal model is too restrictive for the coding stage. With this point in mind, designers will be able to validate a formal model according to the critical elements of specifications, while enabling flexible handling of the non critical elements in the development stage.

2.5.2 Evaluation of Alloy in Pervasive Computing

Using ALLOY has been an efficient way to improve our understanding of the model and check its consistency. It has enabled us to obtain a final functional version of the UML diagrams and to ensure better system reliability. It has also been an approach to increase our confidence that critical requirements are satisfied.

Nevertheless, some limitations exist with ALLOY. First, the more the number of entities generated by the model is large, the more the complexity is high. The ALLOY analyzer imposes a limit on the number of atoms that it can generate to check a model. In other words, the more there are relations between signatures, the more the number of reachable atoms is reduced, thus it is better to avoid large relations. Distributed environments implies large entities generation, and consequently large atoms use. Our implementation avoids this limit by defining relations which are at most ternary ones. For this reason, developers need to gradually construct the ALLOY model to avoid

complex and resource-consuming actions. This progressive design also allows one to detect inconsistency earlier in the development process. Second, as a first-order language, ALLOY has the *frame problem*: the developer has to indicate in an action specification, for each field of an object whether the value of this field is unchanged or modified. This can be costly in terms of development time, especially in the case of dynamic system such as INOVUS. DYNALLOY [36] is an extension of ALLOY to describe properties of execution traces, in the style of dynamic logic. DYNALLOY simplifies the way to represent dynamic systems and allows one to model in the same way as programming (conditions, loops, ...). It is the future candidate to study in the INOVUS project.

2.6 Conclusion

This paper proposes the introduction of model checking to ensure safety for people while developing applications in smart houses. By adding this new step in the development process, the INOVUS project aims to be an alternative and complete technology to the existing solutions in terms of safety in the kitchen for PwSN.

The model checking approach enhances the design of the software under development and helps to ensure satisfaction of requirements. As a consequence, this paper shows how formal specifications can be an asset for pervasive safety applications development in smart environments. The INOVUS design reveals critical steps in which formal specifications improved software design for pervasive applications.

Further work on this project is to extend the range of the INOVUS safety applications in terms of risks contexts and ability of the decision engine to infer user's activities and danger level. This should be done in next increments. To do so, the choice of the model checker has to be optimized because ALLOY does not seem to be the best one for pervasive applications in smart houses. An evaluation of DYNALLOY should be done. This will then bring an implementation in our smart environment, in order to validate the proposed safety solution.

- FIN DE L'ARTICLE -

2.7 Compléments sur l'utilisation du vérificateur de modèle

L'article présenté dans ce chapitre a permis d'exposer l'approche adoptée dans le développement de l'application INOVUS. Dans cette approche, il a été choisi d'introduire l'utilisation des méthodes formelles afin de vérifier le modèle de solution proposé. Nous avons donc fait appel au *model checking*, et plus spécifiquement au vérificateur de modèle ALLOY³ [53]. Cette utilisation a conduit à une représentation du modèle de solution d'INOVUS, puis a permis de s'assurer du respect des exigences critiques de détection et de gestion des dangers liés à l'utilisation de la cuisinière.

Dans la mesure où ces éléments n'ont été que brièvement traités dans l'article précédent, cette section présente succinctement le vérificateur de modèle ALLOY, pour ensuite expliciter son utilisation dans le cadre du projet INOVUS.

2.7.1 Motivation

Les méthodes formelles proposent différents outils permettant de faire de la vérification de modèle, dont ALLOY [52]. Ce *model checker* utilise la logique du premier ordre pour fournir une analyse mathématique sur un modèle donné en entrée. Cette analyse permet de vérifier la cohérence du système décrit. En tant que langage de spécification formelle, ALLOY possède sa propre syntaxe, et permet ainsi de construire une représentation du système à analyser. On identifie de cette manière l'ensemble des entités qui interagissent au sein du système, ainsi que les liens entre elles, et les cardinalités associées à ces relations. Le modèle construit peut ensuite être analysé par le biais de l'outil *Alloy Analyzer*. Cet outil apporte une vérification automatisée de la cohérence du système et du respect de propriétés spécifiques. Ces propriétés correspondent de manière générale aux exigences critiques du système à développer.

Dans le cadre de ces travaux de maîtrise, on souhaite effectuer une preuve de concept de la solution proposée. Pour y parvenir, l'analyse conduite par le vérificateur de modèle doit conclure à une détection efficace des situations à risques, ainsi qu'à une gestion appropriée des réactions fournies. C'est donc la structure du système ainsi que

3. Site web officiel d'Alloy - <http://alloy.mit.edu/>

le raisonnement qui y est associé qui doivent être vérifiés. Ces éléments étant aisément représentables (i.e. sans nécessiter de détails d'implémentation), le développement du modèle formel ne s'est fait qu'à un haut niveau d'abstraction.

L'ensemble de ces éléments, associés à une syntaxe abordable (car proche d'un langage orienté objet), ont mené au choix d'ALLOY parmi les différents outils existants.

2.7.2 Construction du modèle formel

La syntaxe offerte par ALLOY permet de représenter les différentes composantes constituant notre modèle de solution. Il est ainsi possible de modéliser non seulement la structure de notre système, mais aussi son comportement.

Modélisation de la structure

Afin de représenter au mieux la structure de la solution choisie, les différentes entités (appelées *signatures*) interagissant ensemble ont été identifiées. La figure A.1 de la section A.1 à l'annexe A présente ainsi la structure du modèle adopté dans la phase de spécification formelle. Afin de conserver la validité de l'utilisation du *model checking*, cette structure a servi de base à la conception et au raisonnement mis en place lors de la phase d'implémentation du prototype. Cette transition entre la spécification formelle et la phase de codage sera présentée dans le second article sur le projet (cf. chapitre 3).

Modélisation du comportement

Une fois les différentes entités du modèle mises en place, il a été nécessaire de représenter les interactions possibles entre elles compte-tenu de leur relations. En effet, la structure fournie par l'ensemble de signatures ne présente qu'une image statique. Cette image correspond à un état générique du système. Afin d'intégrer une notion dynamique à notre système et ainsi faire évoluer les différents états possibles au cours du temps, il est nécessaire de représenter les transitions possibles d'un état à un autre. Cette évolution temporelle du système est représentée au sein d'ALLOY par le biais de *prédicats*, qui permettent d'indiquer les modifications entre les ensembles, sous des

2.7. COMPLÉMENTS SUR L'UTILISATION DU VÉRIFICATEUR DE MODÈLE

pré-conditions et post-conditions données.

La dynamique ainsi introduite au sein du système permet alors de modéliser les différents éléments nécessaires à la validation du modèle : évolution de l'environnement, actions de l'utilisateur, réaction du système face à un contexte donné (e.g. déplacements de l'utilisateur, démarrage d'un élément de cuisson, partage d'une information par le système si un danger est présent, etc...).

2.7.3 Phase de vérification

À partir du modèle formel établi, la phase de vérification a pu être initiée afin de s'assurer qu'il soit réalisable, et qu'il réponde aux attentes. Comme expliqué précédemment, cette vérification se fait aussi bien sur la cohérence du système que sur le respect de propriétés spécifiques.

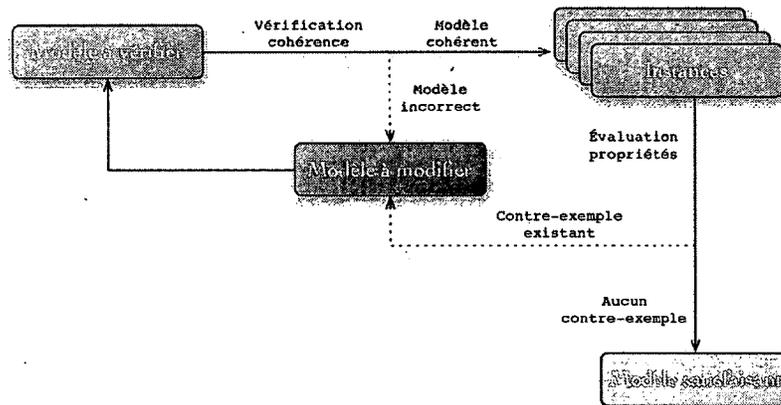


Figure 2.3 – Processus de vérification formelle d'un modèle avec ALLOY.

La figure 2.3 représente le processus de vérification conduit par *Alloy Analyzer* sur un modèle donné. Pour qu'un modèle soit considéré comme satisfaisant, il faut qu'il complète l'ensemble du processus. Il faut donc que *Alloy Analyzer* soit en mesure d'en trouver une instance, mais aussi que cette instance respecte avec succès l'ensemble des propriétés fixées. On remarque sur la figure 2.3 qu'un cycle se forme dès lors qu'aucune instance du modèle ne peut être générée, ou qu'il subsistent des propriétés non vérifiées. Dans le cas du projet INOVUS, neuf itérations ont été nécessaires pour aboutir à une version du code ALLOY qui permette de générer effectivement

CHAPITRE 2. APPROCHE DE CONCEPTION LOGICIELLE

une instance satisfaisante du modèle. Globalement, les erreurs détectées ont été dues à une mauvaise définition des configurations représentant un risque pour l'utilisateur. Néanmoins certaines versions infructueuses du modèle ont fait ressortir des erreurs dans les enchaînements possibles d'événements : certains prédicats apparaissaient réalisables alors qu'ils ne devaient pas l'être.

Il est à noter de plus que, dans notre cas, les itérations successives de ce processus peuvent être expliquées par le caractère expérimental de cette approche.

Ainsi, l'utilisation du *model checker* pour assurer la viabilité du modèle a dans le même temps permis d'améliorer l'exactitude des algorithmes de détection des configurations à risques, tout en se basant sur une représentation valide du monde réel.

Enfin, la figure 2.2 présente le processus de développement logiciel classique, et indique le positionnement de l'étape additionnelle concernant la spécification formelle. Il est important de préciser que cette étape supplémentaire ne vient pas surcharger le développement classique, mais s'avère complémentaire : l'anticipation des vérifications apportées permet d'éviter des erreurs de conception, et donc de réduire la phase de tests, tout en augmentant leur couverture.

L'approche de développement proposée ici se positionne donc comme une version renforcée du processus classique.

2.7.4 Code Alloy produit

Des exemples du code ALLOY pour ce prototype d'INOVUS sont présentés à l'annexe A. La section A.1 présente successivement le code de certaines signatures du modèle, puis de certains prédicats, puis enfin du fichier d'exécution de la vérification par l'analyseur. Par ailleurs, la section A.2 de cette annexe fournit un exemple de propriété en ALLOY, ainsi que la liste des propriétés qui ont été établies.

Le but de ces extraits de code est de faire ressortir efficacement le rôle de chacune des étapes de la spécification formelle réalisée.

D'autre part, l'intégralité du code ALLOY développé dans le cadre de ces travaux est disponible sur l'intranet du laboratoire Domus.

Chapitre 3

Validation d'approche par le développement d'un prototype d'Inovus

Ce chapitre correspond au second article présentant le projet INOVUS. Cette version de la publication n'est pas encore définitive puisque le journal pour lequel elle est destinée n'a pas encore été défini.

Dans la continuité de l'article présenté au chapitre 2, ce chapitre propose une validation de l'approche de développement proposée. Il présente ainsi la phase d'implémentation d'un prototype d'INOVUS, et les résultats qui en découlent.

Ce second papier étant destiné à être compréhensible pour un lecteur n'ayant pas la connaissance du contenu de l'article présenté au chapitre précédent, les éléments essentiels y sont à nouveau brièvement présentés. Il s'agit donc d'une version étendue de l'article précédent. De ce fait, les trois premières sections s'avèrent être une synthèse des chapitres 1 et 2.

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

Contribution de l'étudiant

L'étudiant *Thibault de Champs* est premier auteur de cet article, puisqu'il en a été le principal contributeur. Les différentes révisions de l'article ont été menées avec les superviseurs de l'étudiant, Bessam Abdulrazak & Hélène Pigot, qui constituent donc les auteurs additionnels.

Versions antérieures

Il s'agit de la première version de cet article, élaborée dans le cadre de la rédaction de ce mémoire.

Inovus: A Pervasive Application to Guarantee User Safety while Realizing Cooking Activities using the Kitchen Stove of a Smart Home

Thibault de Champs & Bessam Abdulrazak & H el ene Pigot
Laboratoire DOMUS
D epartement d'informatique, Universit e de Sherbrooke,
Sherbrooke, Qu ebec, Canada J1K 2R1

Keywords: smart home; pervasive computing; home safety; assistive technology; cooking activities; oven; ubiquitous computing; healthcare; kitchen; OSGi;

Abstract

By nature, the human home aims to be a safe and accommodating place. Nevertheless, numerous activities of daily living may threaten user safety. Among the hazardous tasks, this paper specially tackles cooking activities. Indeed, the kitchen stove use appears as a major cause for burns, fires, and gas/smoke intoxication. People suffering from cognitive impairments are mainly concerned about these risks, leading to providing them a safe home support.

Based on a state-of-the-art on the current assistive technologies for ensuring user safety in the kitchen, this paper presents the software layer of the INOVUS project, which aims to be a smart-home-based safety solution designed for people with special needs while using the stove. In order to assure reliability and efficiency for this pervasive application, we present here the formal specification we applied using the ALLOY model checker, thus improving our software design. This paper presents the OSGi implementation of this application. This work results in a functional and deployable prototype in a smart environment such as the Domus Lab smart apartment.

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

3.1 Introduction

Healthcare for *People with Special Needs* (PwSN¹) represents a substantially addressed research area. Among the impairments the PwSN present, the cognitive ones may provoke numerous obstacles while realizing *Activities of Daily Living* (ADL) [40]. As a result, assistive devices are spreading to help them in fulfilling these basic needs [10, 41, 58, 68], and their purpose regularly intends to act as cognitive orthosis. The assistive devices can thus definitely be used in PwSN home care, to comply with the crucial role of providing reminders to the habitants and support in tasks achievement of their daily living routine.

Moreover, while taking care of an impaired human being at home the caregiver oftentimes experiences burdenhood [26]. For this reason, the aforementioned assistive devices are also used to alleviate this workload. However, despite the proven efficiency of these implements, they can not substitute caregivers in all ways, especially concerning safety which is a critical consideration in home support.

Safety is a major issue for a single person at home. Multiple efforts have been conducted in many fields such as gerontology, occupational therapy, psychology, or emergency services in order to identify risks inside the residence, and their sources [5, 22, 29, 39, 47]. They can be classified in four major physical risks, which are falls, burns, fires, and smoke/gas intoxications.

Falls appear to be the most addressed threat in the home, whereas burns and fires are partially addressed. This hence motivates the work presented in this paper.

Many studies [47] identify three categories of injuries concerning burns (scald, flame, or electric) mainly occurring in the bathroom and the kitchen, with established common appearance contexts based on stove use [30].

Regarding fires, several sources are presented, but the highest risks arise while smoking or cooking [29]. Again, studies [43] specifically highlight the oven-related tasks as the first fires cause.

The literature discussing respiratory poisoning at home seems succinct. Many works take into account the carbon monoxide hazard [78]. Nevertheless, some studies sug-

1. People with Special Needs: Elderly or people with disabilities.

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

gest other intoxication causes such as cooking activities. However this context seems to be regarded as a consequence of fire [5, 55].

In addition to these risk situations, some authors suggest worsening factors for inhabitant vulnerability. For instance, Hill *et al.* in [47] show the prevalence of alcohol and cigarette consumption as increasing elements in domestic accidents. Dementia, and more widely cognitive impairments, are also pointed out in these factors [5, 29, 47].

Hence, oven use appears to represent a critical activity context for user safety. For this reason, some solutions are proposed to prevent and detect accident. Their intention generally aims for a common use and does not have relevance for a specific population such as PwSN or elderly people. These considerations reveal the need for a technological solution specifically tailored for PwSN in home support in order to ensure their safety while using the stove.

Existing solutions for detecting or preventing domestic accidents present some limitations: feedback features are not adapted to PwSN who present altered planning and judgment abilities. Therefore, this paper proposes a novel concept for managing range-related dangers through the use of a smart home. This constitutes the purpose of the INOVUS project.

INOVUS aims to be an integrated and powerful technology by offering an alternative to the existing products. Nevertheless, as a critical system, INOVUS implies highly reliable constraints. Consequently, we chose to adapt the software development process [23]. This paper now presents the implementation we conduct for the INOVUS system to validate this approach.

This paper is structured as follows: Section 3.2 presents actual existing solutions in terms of risks management at home and related emerging perspectives. Section 3.3 presents the INOVUS software through the proposed model, its reasoning process, and the formal verification applied on the developed model. Section 3.4 presents the development through its architecture and code implementation. Section 3.5 then describes the validation process of the developed application. Section 3.6 finally discusses the results, and Section 3.7 concludes on the proposed solution.

3.2. HOME SAFETY

3.2 Home Safety

As maintaining PwSN at home becomes an emerging common situation, home safety appears to be a progressively addressed issue and related works are spreading. Safety assistance may take several forms, as well in terms of prevention as technological products. This section presents existing solutions to the home safety issue, and will introduce the purpose of the INOVUS project through the use of smart homes.

3.2.1 Home Safety Evaluations

According to the extensive home support development, people awareness toward hazards surrounding them in their house increases. This improvement can be mainly explained by the work of organizations on the sensitization of home-alone PwSN and their caregivers (e.g. Alzheimer Society² [83]). Regrouping caregivers, clinicians, and even insurers, these organizations share information in order to prevent home hazards and improve the quality of life of the impaired person.

Furthermore, this prevention is also initiated by clinical studies in occupational therapy, which aim to provide home evaluation of environmental threats. For instance, Anemaet *et al.* [7] suggested to conduct an accurate examination of each elements constituting the home environment, in order to record the at-risk areas. To carry out this inspection, they proposed to use the *Functional Environment Assessment* tool, and the *Objective Safe at Home* tool, and to follow standards from the *Joint Commission on Accreditation of Healthcare Organizations*³ [73].

Following the same objective, other tools were developed, such as the *Safety Assessment of Function and the Environment for Rehabilitation* (SAFER Tool [57]), the *Westmead Home Safety Assessment* (WeHSA [21]), or the *Home Falls and Accidents Screening Tool* (HOME FAST [62]).

These tools aim to highlight risk factors surrounding the home inhabitant. Nevertheless, they essentially represent prevention methods and cannot completely avoid domestic accidents. Therefore, additional answers to the home hazards are needed

2. Alzheimer Society UK - Remaining Independent - <http://www.alzheimers.org.uk/site/scripts/documents.php?categoryID=200349>

3. Joint Commission - <http://www.jointcommission.org/>

in order to react when a casualty occurs, and this assistance is mainly provided by using technological products.

3.2.2 Current Technological Coverage for Home Safety

Numerous assistive devices are offered to help the user to safely realize ADLs. Some studies [10, 41, 58, 68] present a state of the art of these technologies in terms of nature, maturity, development process, or functionalities. Most of them are physiological sensors, bio-sensors, or cognitive orthosis, specifically designed for monitoring data about user health (e.g. blood pressure, heart rate,...) or to provide them reminders or assistance in activity realization. Thus, these devices allow to follow a person at home.

Nevertheless, among these solutions, some of them are specifically proposed to ensure dweller safety. The intention of these apparatuses is, for the most part, targeted on a specific jeopardy. For instance, various instruments are proposed for fall detection, such as embedded sensors on clothes [10], tele-health solutions (e.g. EDAO⁴ [82]), or travel helps. Smoke/gas intoxications and fires are addressed by widespread sensors. Finally, burns are handled with temperature indicators or limiters⁵ [37].

Regarding the oven use, numerous studies identify burns, fires, and smoke intoxications as the three major physical threats for the home inhabitant in range-related contexts. Few technologies are already offered to manage these risks, and are summarized in Table 3.1. Among the offered solutions, we distinguish the widespread technologies, such as smoke sensors, the integrated technologies, such as hot surface lights, and the specific technologies that are added on a conventional oven. For instance, devices such as Stove Guard⁶ [59] or SafeCook⁷ [50] are available if you do not have range-integrated timers to stop the cooking elements after a predetermined duration. These additional devices may prevent a fire or an intense smoke emission when forgetting the meal while cooking it. Another appropriate reaction in this hazardous context could be to switch off the power/gas supply of the stove, as offered by

4. EDAO - <http://www.edao.com/domicile/edao-system/>

5. *Plumbing to Prevent Domestic Hot Water Scalds* - <http://www.hgexperts.com/article.asp?id=5135>

6. Stove Guard - <http://stoveguard.ca/>

7. SafeCook - <http://www.safecook.ca/>

3.2. HOME SAFETY

Table 3.1: Management technologies for range-related threats.

Technology	Product example	Covered risk
Smoke/gas sensors	<i>Widespread</i>	Fire, intoxication
Auto-stop timers	<i>Stove-integrated functionality</i> Stove Guard SafeCook	Fire, intoxication
Light indicator	<i>Stove-integrated functionality</i>	Burn
Power/gas supply manager	SafeCook HomeSenser	Fire, burn, intoxication
Automatic extinguishers	Stove Top FireStop	Fire

the HomeSenser⁸ [31]. Finally, automated extinguishers (e.g. Stove Top FireStop⁹ [32]) which spontaneously deliver a suitable agent, such as carbon dioxide, when a fire starts can also be added on the stove top to interrupt fire propagation, thus avoiding accidents.

These auxiliary systems extend the risks prevention coverage, and their effectiveness has been proven [29]. Nevertheless, they present some limitations that reduce the resulting impact. First, they need to be installed as extra features on the existing home cooking installation. As they are separated artifacts, each one must be regularly checked to be fully effective. Consequently, the resulting requirements on maintenance represent an additional concern for the user. Moreover, timers and power supply managers have to be set up leading to an extra action when preparing a meal. It implies to master these technologies and this can represent some difficulties for the cognitively impaired people. Finally, configuration parameters are often restrictive. As a consequence, contextualization features toward the user needs are thus reduced. Similarly, user interactions are restricted to the device-embedded elements, and may not fit with the user impairments.

To answer these issues, an emerging approach consists to use smart environments.

8. HomeSensers - <http://www.homesensers.com/>

9. Stove Top FireStop - <http://www.stovetopfirestop.com/>

3.2.3 Smart Homes

Numerous research groups are currently working on the development of smart homes (e.g. Domus Lab¹⁰ [25], The Gator Tech Smart House¹¹ [69], or The Aware Home¹² [72]). Such environments are built with fully integrated high technologies, thus allowing the use of ambient intelligence to provide assistive home-support and tele-care services to the inhabitant. Consequently, safety-related projects are spreading in such environments.

As we are developing home-support technologies for PwSN, INOVUS perspectives will offer a home safety solution which must be as transparent to the user as possible. Consequently, the use of smart homes appears to be an adapted solution to the safety problem.

3.3 Inovus Concept

This section presents the INOVUS model. Based on smart home technologies, this project aims to offer a pervasive support to ensure safety in cooking-related contexts for an inhabitant presenting cognitive impairments. The INOVUS concept, the associated model, and the resulting formal analysis, are explained in this section. These software development steps have previously been presented in [23].

3.3.1 Inovus purposes

The Domus Lab Smart Home consists to an apartment equipped with sensors (e.g. infrared sensors, microphones) and effectors/actuators (e.g. screens, speakers) networks. These equipments are distributed in each room, in a fully integrated and transparent way, thus allowing to sense home activity and offering user interaction.

Our proposition to address cooking safety issues for a home-alone PwSN is to use these infrastructures in order to detect elements presenting potential hazards for the inhabitant. Indeed, according to the house-provided sensors, the introduction

10. <http://domus.usherbrooke.ca/>

11. <http://www.icta.ufl.edu/gt.htm>

12. <http://awarehome.imtc.gatech.edu/>

3.3. INOVUS CONCEPT

of pervasive intelligence allows to record a specifically chosen inputs set. Combined with data fusion, such a process leads to real-time house state monitoring. Artificial intelligence techniques then offer to rule between the at-risk or safe state for the current context.

Moreover, the actuators network proposes a wide range of possibilities for human-machine interaction, and thus offers, for each defined risk situation, to adapt the provided information and reactions according to the user needs.

In order to illustrate this concept, we examine the following scenario.

Situation of Risk Mr. Smith is a 74-year-old man who lives alone. One year ago, he was diagnosed with Alzheimer disease in the early stage. To facilitate his daily living, his daughter visits him everyday at noon to prepare lunch and dinner meals for him. Two months ago, as his daughter was late, Mr. Smith decided to prepare food using the stove. Few minutes later, he moved to the living room to watch his favorite TV show, forgetting the meal on the stove. When his daughter arrived forty minutes later, she noticed a dense smoke in the kitchen and her father was hardly breathing.

Inovus intervention To help Mr. Smith and his daughter, the INOVUS intelligent oven would have pro-actively replied to this situation. (1) Detecting that Mr. Smith was for too long outside the kitchen , INOVUS would have plan to send him a warning. Noticing that he is watching his program, the message will be sent on TV or through speakers of the smart environment. If Mr. Smith remains watching TV, INOVUS would automatically shut down the oven. (2) Meanwhile, presence of smoke is detected. This critical risk which threatened Mr. Smith would result in an emergency stop of the oven. In addition, his daughter or/and relevant emergency services is informed about the situation. Such a solution would ensure safety of Mr. Smith and serenity for his daughter.

3.3.2 Inovus Architecture

The assistance offered to Mr. Smith requires to detect the risk situations, analyze their potential dangers, and generate feedback to prevent these dangers. For this

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

purpose, INOVUS is based on the distributed sensors and actuators in the environment, and on user's requests tracking, to infer about the current risk level linked to the stove use and plan assistive actions to avoid potential accidents. INOVUS encompasses these inference steps inside a data fusion module, a reasoning module, and a management module (Figure 3.1). More details about this architecture are given in [23].

According to the risk situation, INOVUS will either send an emergency call to the caregivers, or a control command to shut down the stove elements, either a warning message to the user in order to help him resume the risk.

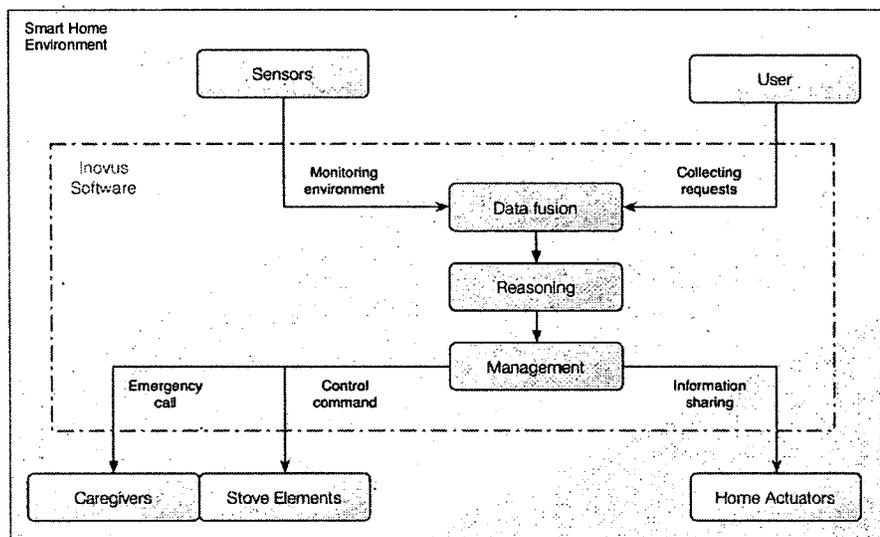


Figure 3.1: INOVUS architecture as home safety solution.

As soon as the inhabitant uses the stove, INOVUS becomes in an alert mode. While the inhabitant uses adequately the stove, INOVUS keeps silent but still analyses the situation to ensure no risk is present. Such follow-up is necessary to detect risks at the beginning. However, the follow-up do not prevent the inhabitant to use the stove. The aim is also to encourage the user to be autonomous using as much as possible its cognitives functions.

When the reasoning engine infers a critical risk situation, then it will launch emergency reactions: (1) take control of the in-use stove elements, (2) inform caregivers according to the importance of the threat. These two emergency reactions should occur when the reasoning engine considers that a previous shared information has

3.3. INOVUS CONCEPT

not been received by the user.

Therefore, INOVUS embeds and automatically manages three running mode: the silent follow-up, the assistive mode, and the emergency mode.

As the INOVUS application embeds elements of activity recognition, data fusion, and decision making, artificial intelligence techniques appear to be an appropriate solution to the software reasoning part. Therefore, neural networks, fuzzy logic, or other techniques may offer extremely powerful answers for a given input, but since this first INOVUS prototype aims to be a proof of concept of a smart-home-based safety solution, we simplified the reasoning engine to a rule-based system.

For this purpose, we first formalized our targeted risk coverage by selecting few opportune contexts for fires, burns and intoxication [23]. Based on the available sensors in the Domus Lab Smart Apartment (i.e. the quantifiable environmental factors), each context was then characterized by a set of data, thus constituting a digital representation for each included threat. An information strategy and a reaction strategy were additionally associated for each risk situation description. The chosen reasoning logic was then to conduct a real-time observation of the home representation. Each predefined state is compared to the current one, thereby launching management strategies in case of correspondence.

Although this artificial intelligence approach appears very simple, it offers satisfying results for a proof of concept. Nevertheless, the processing algorithm will obviously be improved in further software increments, with more flexible and efficient techniques.

3.3.3 Model Checking in Inovus development process

Developing assistive technologies naturally implies human concerns. This assertion is reinforced while dealing with PwSN. As a safety-related technology, INOVUS must prevent the home occupant from injuries. As a consequence, such a system requires high level reliability constraints. However, the literature demonstrates that a major cause of failure for a specific software lies in its design. Consequently, in order to maximize INOVUS productiveness, our proposition consists to the introduction of formal specifications as an additional step in the development process of a

smart home-related software [23]. We thus presented a formal model of the INOVUS solution, using the ALLOY model checker. Main elements of this usage are presented here, whereas the entire INOVUS development process is exposed in [23].

Formal Modeling

ALLOY [52] is a specification language used to express complex structural and behavioral constraints in software design. It allows to build a mathematical first-order-logic representation of a software model. Its associated analysis tool, *Alloy Analyzer*, offers automated checking for model consistency and correctness according to selected properties.

Building the INOVUS ALLOY representation requires identifying software key entities. For this purpose, we needed to specify the elements generating interactions, first between INOVUS components, and then with the host system (i.e. the home environment and the user). We thus built a high level UML representation of the model presented in Figure 3.1. The resulting diagrams were next transposed in ALLOY syntax, which offers features for representing static and dynamic parts of the INOVUS system.

Checking Process

Alloy Analyzer was previously mentioned as the model checker analysis feature. Figure 3.2 specifies the verification methodology for a given ALLOY representation.

We apply the analysis process to the INOVUS formal model. By checking the proposed structure and the possible interactions between entities, *Alloy Analyzer* highlights inconsistencies, leading to several adjustments. In our case, nine iterations on the INOVUS model were necessary to reach satisfying instances under various scopes, hence improving as well interactions with the environment as detection algorithms.

In addition to consistency checking, *Alloy Analyzer* offers features to ensure compliance with specific requirements. These requirements are formulated in ALLOY syntax as assertions. The analyzer evaluates each property on the found model instance. As INOVUS critical points are reliability in hazardous contexts detection and adapted retroactions, we formalize these purposes into ALLOY properties to be evaluated.

3.3. INOVUS CONCEPT

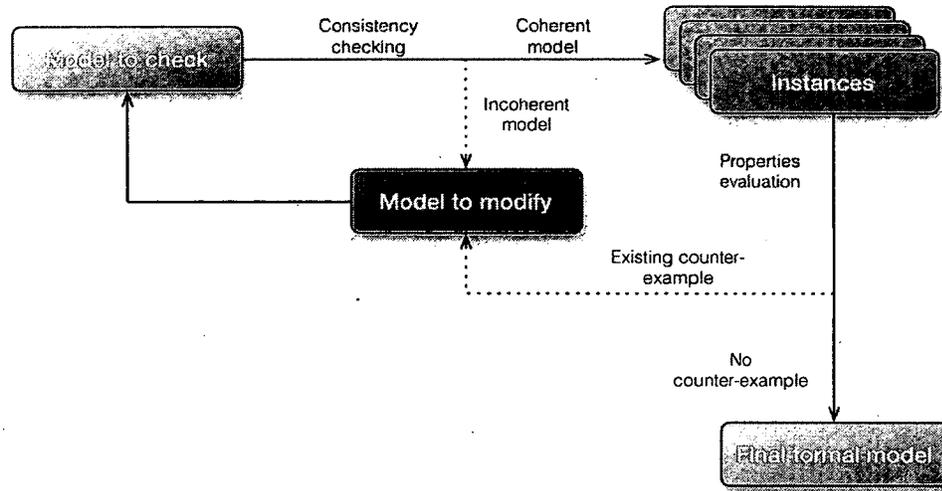


Figure 3.2: ALLOY formal verification process.

3.3.4 From Formal Specification to Code Implementation

The checking step ensures that the INOVUS model fulfills the requirements. The next step is to produce a code implementation based on the ALLOY representation. Nevertheless, some elements must be taken in account in the software development phase, in order to fully integrate the benefits from the formal specification phase.

Indeed, an appropriate formal specification lies in a correct abstraction level of the system to model. It isolates its critical elements according to specific requirements. Concerning INOVUS, the formal model is a high level representation of the system since its critical elements stands on the reasoning logic and the information flow regarding a risk management. The ALLOY specification thus does not represent a single implementation, though a possible one. The functional code implementation therefore does not require to strictly follow the ALLOY implementation, however the reasoning logic and the associated high level data flow must be preserved.

Next sections present the software implementation and validation, which result in a functional INOVUS prototype.

3.4 Software Implementation

Section 3.3 presented the INOVUS model of solution and the associated formal verification with model checking. In order to carry out a satisfying INOVUS prototype, a Java implementation has been developed and validated using the Domus Lab resources. However, the implementation design must respect several requirements. Firstly, it has to integrate the previously developed ALLOY model, hence ensuring a successful reasoning. Secondly, some autonomy, flexibility, maintainability, and contextualization factors must be considered. The following section thus presents the system architecture and the corresponding code structure.

3.4.1 System Architecture

A crucial INOVUS requirement is the application independence toward the targeted system. This is motivated by the fact that INOVUS has to be deployable regardless of the client system. For this purpose, we adopted a 3-layers architecture (Figure 3.3). According to its purpose, each layer implies a specific data flow for exchanges.

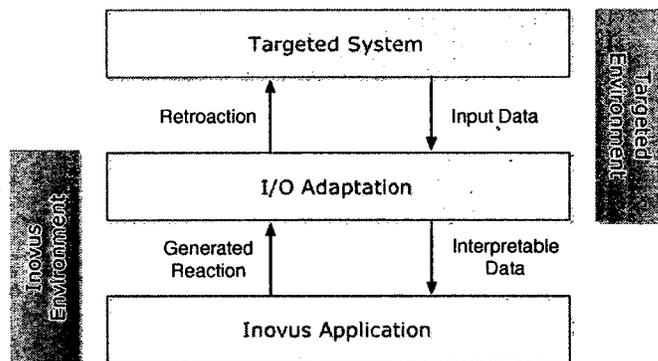


Figure 3.3: INOVUS layers architecture and the resulting data flow.

In order to improve the autonomy of the proposed solution, the INOVUS application encompasses its own environment description. Figure 3.3 thus specifies for each layer its environment knowledge, and its Input/Output (I/O) interactions.

The top layer represents the smart environment where our solution is deployed. This layer provides the input data to be processed for risks contexts detection. To be

3.4. SOFTWARE IMPLEMENTATION

able to reason on the events detected in the environment, the information provided from sensors need to be translated to be interpretable by INOVUS. This is the purpose of the adaptation layer. Therefore, the first two layers have clear representation about the targeted environment, whereas the INOVUS layer does not have. Based on its own environment knowledge, the INOVUS layer processes the adapted input data and provides reactions when necessary. Information to the user and stove control commands are then forwarded to the client environment, after being correctly formatted by the intermediate layer.

Such an architecture appears to be the simplest one to ensure continuity of the INOVUS layer services. Nevertheless, an important drawback then also emerges: the adaptation rules must re-written in line with the client system description. To simplify this process, we adopt a component-oriented programming approach, hence allowing to focus on the key functionalities. This choice also outcomes in a greater integration of the programming requirements previously exposed.

3.4.2 Code Architecture

We formerly introduce flexibility, maintainability, and contextualization constraints for the design of the INOVUS code implementation.

First, the application must cover an evolutionary risk situations range. Actually, this coverage has to be easily adapted to the user's daily living routine. As a consequence the application should allow to perfectly match the inhabitant's needs by authorizing the introduction of new hazardous contexts.

Moreover, in order to be as much flexible and maintainable as possible, it appears obvious to isolate the critical code elements (e.g. reasoning engine, data description, or interaction features). In this manner, some modifications can be smoothly initiated, without altering other functionalities.

These considerations reinforce the relevancy of the component-oriented programming previously mentioned. Hence, the technological choice we made was to use the *Open Service Gateway Initiative*¹³(OSGi) framework [6]. The OSGi framework is a module system and service platform for the Java program-

13. OSGi Alliance - <http://www.osgi.org/>

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

ming language that implements a complete and dynamic component model. It allows to develop a set of bundles (i.e. components) and offers features to manage services between them. Each bundle implements a specific functionality, and can be either a provider and/or a consumer of services. The resulting code architecture thus affords splitting the main functionalities in separate bundles for a software.

Referring to the INOVUS application, the independence, modularity, contextualization, and maintainability constraints are addressed by the architecture presented in Figure 3.4. Inspired by the UML standard, this figure gives the code organization in components and their associated interfaces. We define inside the two INOVUS-related layers (i.e. INOVUS itself and the adaptation layer) which functionalities have to be isolated from the rest of the code. This leads to the development of seven bundles for our safety solution, and two bundles for the adaptation layer.

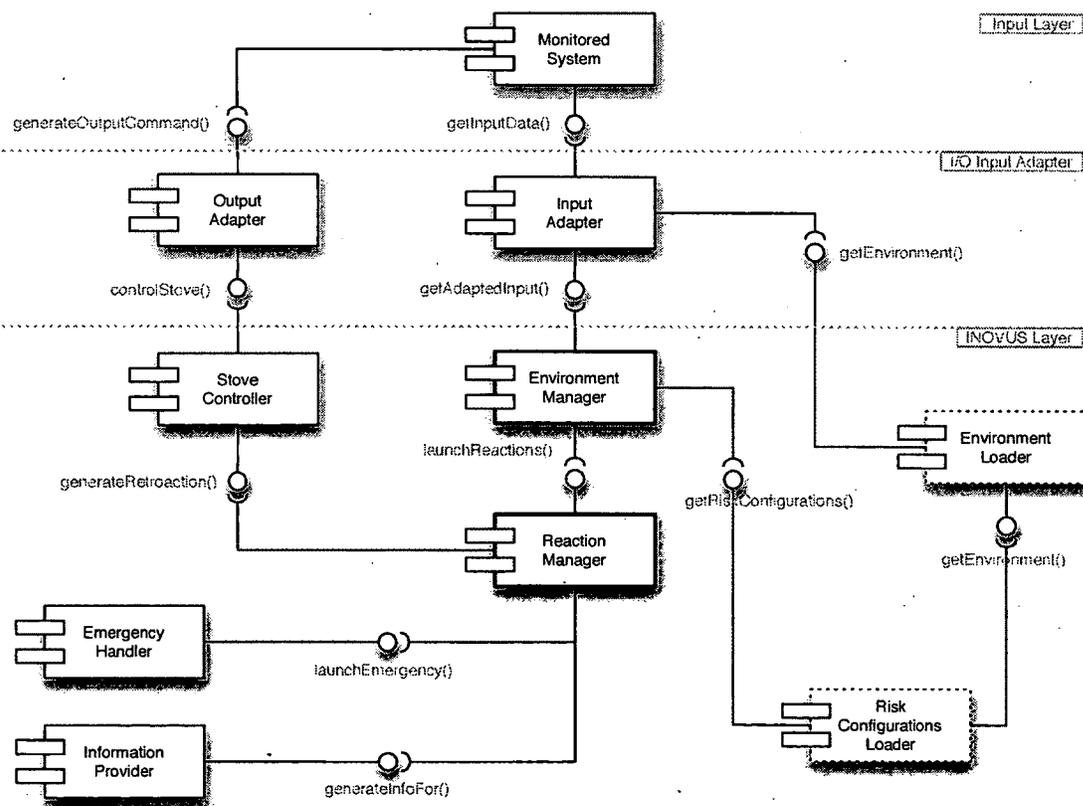


Figure 3.4: INOVUS components diagram.

3.4. SOFTWARE IMPLEMENTATION

The proposed bundles can be classified in four categories: reasoning bundles (bold frame), description bundles (dotted frame), interaction bundles, and adaptation bundles. The purpose of each one is described below.

Description Bundles

We previously presented in Figure 3.3 the 3-layers architecture, with their respective environment knowledge. We explained that the INOVUS layer has its own environment description. This description is provided throughout an XML file, which details the available kitchen stove elements, and for each room, the integrated technologies (i.e. sensors, actuators, and their possible states). As a consequence, the purpose of the *Environment Loader* bundle is to read this file, and translate it into Java objects, thus providing an interpretable Java representation of the home environment. This representation is then used by the *Input Adapter* and the *Risk Configurations Loader*, respectively in order to adapt the input data into understandable ones for the INOVUS layer, and for building a Java representation for each covered risk situation.

A similar role is assumed by the *Risk Configurations Loader* bundle: each risk configuration is described in an XML source file, which contains the sensors set (and their critical values) used for detection. This description is provided in agreement with the loaded environment. The emerging Java representation is used by the reasoning engine for hazard detection. This representation comes with information and reaction strategies, which specify how to inform the user or to react, according to the current context.

Reasoning Bundles

The reasoning logic described in Section 3.3 is implemented within two bundles: the *Environment Manager* and the *Reaction Manager*.

The first one calls the *Input Adapter* service following a predetermined frequency, and processes the comparison between the current environment state, and each loaded risk configuration. The detected hazardous contexts are handled by the *Environment Manager* in order to initiate reactions management throughout the *Reaction Manager* service.

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

Concerning the *Reaction Manager*, when its service is consumed, it launches the information and reaction strategies for each provided hazardous context. To process the risk management strategies, this component calls services from interaction bundles.

Interaction Bundles

The remaining components in the INOVUS layer supply functionalities for providing feedbacks to the inhabitant. The *Information Provider* shares information to the user about a potential danger using the home actuators. According to the reaction strategy associated to the concerned risk situation, the retroaction and emergency services are called when needed, hence ensuring an efficient support: the *Emergency Handler* bundle offers features for launching emergency call to caregivers or relevant services, and the *Stove Controller* bundle initiates control on active stove elements in order to eliminate the hazard source. Calls for these services are sequenced with the reaction strategy of each included risk situation.

Adaptation Bundles

The last two components constitute the adaptation layer. Input and output data for the INOVUS layer must be adapted in order to respectively be understandable by the reasoning engine and the targeted system. To do so, the *Input Adapter* bundle transforms the environment data into interpretable ones for the INOVUS layer. Similarly, the *Output Adapter* bundle adapts the control command from the *Stove Controller* in order to effectively transmit it to the client system.

3.4.3 Typical Running Sequence

In order to illustrate the analysis conducted by the INOVUS software on the client system, the following figures present sequence diagrams for monitoring and reaction processes.

Figure 3.5 presents the monitoring process conducted by the INOVUS layer: a typical analysis cycle pulls input data from the client system to the *Environment Manager* through the *Input Adapter*. The obtained data are then processed by the

3.4. SOFTWARE IMPLEMENTATION

Environment Manager for risk situations detection.

This process is reiterated following a predetermined frequency, thus ensuring a continuous monitoring of the home environment.

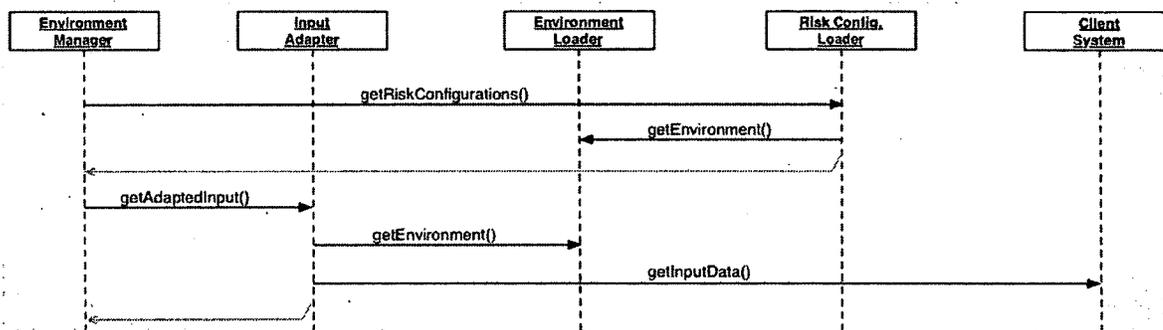


Figure 3.5: The INOVUS monitoring sequencing.

Furthermore, if an input data set inspection results in risks identification, services from reaction bundles are then called in order to manage these hazards. Figure 3.6 proposes a possible reaction sequence in such a case. The reaction process is started for each identified risk situation. The example in Figure 3.6 implies the use of all reaction bundles, nevertheless these steps depend on the reaction strategy attached to the concerned risk context. As a consequence, other possible sequences may only involve the *Information Provider* bundle. Also, the use of the *Emergency Handler* bundle is not mandatory.

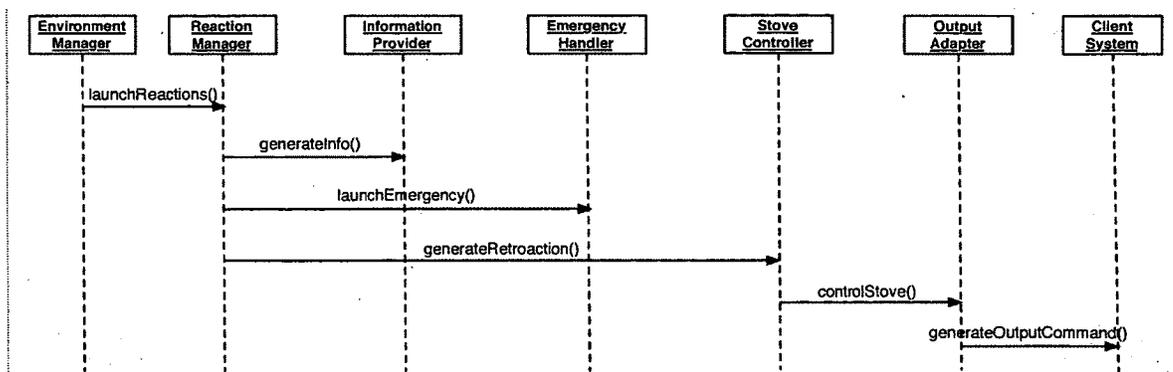


Figure 3.6: A possible INOVUS reaction services sequencing.

3.5 Software Validation

Previous sections detailed the INOVUS project, our approach in developing such a software, and the resulting prototype implementation. This section now presents the adopted methodology and tools to validate our solution.

3.5.1 Validation Methodology

Validation of the produced software requires the following elements after being deployed: efficient and continuous monitoring for the client system, correct hazardous contexts detection, and reliable reaction management for each found risk situation. To reach these goals, the resulting architecture must perfectly implement the reasoning logic formerly introduced by the formal verification with ALLOY. To do so, and in addition to the model checking process and the unitary testing process in design and programming phases, the produced functional prototype needs to be validated with suitable input data in order to be successfully integrated in its execution environment.

However, the Domus laboratory does not integrate all the required technologies for environment monitoring in stove-related contexts. We then choose to validate INOVUS thanks to an ADL simulator previously developed at Domus lab [14]. This simulator substitutes to the real environment by creating a virtual one, and produces sensors outputs through the simulation of predefined activity scenarios. We thus used this simulator to generate sets of input data for representative hazardous situations. Simulated sensors outputs are handled by the INOVUS prototype after being transformed by the adaptation layer, hence allowing software validation while running.

3.5.2 Activity Simulator

Busnel *et al.* [14] propose the use of their activity simulator in order to represent a smart environment and to use it as a basis for simulating the execution of ADL scenarios. In the INOVUS context, such a software contributes on two critical points. Firstly, it tackles the lack of stove-related technologies for cooking activities monitoring in our smart apartment. Secondly, it simplifies the software validation process by

3.5. SOFTWARE VALIDATION

providing simulated data, thus avoiding the generation of real hazardous situations inside the Domus smart apartment. As a consequence, the activity simulator appears to be a suitable solution for the INOVUS software validation.

Simulator Overview

The proposed simulator is specifically designed for a use in smart environments context, and offers features for Activities of Daily Living (ADL) simulation.

Based on object-oriented Petri networks, the simulator gives functionalities for representing a smart environment following four axis: spatial areas, parameters (i.e. sensors), resources (i.e. appliances, objects, food), and possible activities. An activity can be either a task to realize (i.e. user dependent) or an interruption (i.e. user independent), and can be merged with others to create a scenario. Moreover, an activity can involve resources consumption, and parameters evolution.

When launched, the ADL simulator executes the input scenario, and automatically manages interruptions, parameters states, and resources availability.

Simulator Use for Inovus Software

According to the layers presented in Figure 3.3, the ADL simulator acts as the monitored system. To include it inside the OSGi architecture, the ADL simulator has been wrapped up in a bundle. Services were also developed to supply the adaptation layer with simulated data. Thus, each time a transition is completed in the simulation process, parameters states are updated and pushed to a data provider service. These brief modifications allow to connect the simulator to the INOVUS prototype through this data provider service.

In addition, the simulator gives flexibility for the environment representation: all required sensors can be integrated, thus avoiding technological issues that may appear in the smart apartment. All factors contributing to a risk situation can in this way be sensed, and integrated to the INOVUS inspection.

Moreover, the ADL simulator offers another formerly mentioned crucial benefit: dangerous situations for the user can be reproduced without generating such situations directly in the real world. Besides, simulation scenarios were developed to

represent potential risk situations related to the kitchen stove use for an home-alone user in home caregiving. Burns, fires, or intoxication contexts were thus virtually generated.

Furthermore, due to the simulation theory used to develop the ADL Simulator, many minor functionalities could be used. For instance, as the simulator allows to manage interruptions for the running scenario, environmental events can easily be borrowed (e.g. smoke apparition in the kitchen). These features may also be adapted to serve as retroaction realizations for a scenario which implies taking control of in-use stove elements.

3.6 Validation Results

Sections 3.3, 3.4 and 3.5 put forward the INOVUS project and its advancement under the context and the state of the art given in Sections 3.1 and 3.2. This section then presents and discusses the obtained results from the INOVUS prototype.

The previous section introduced the simulator use for generating sensors outputs from predetermined ADL scenarios. In the INOVUS project, these scenarios are used to model the occurrence of dangerous situations while using the kitchen stove. These scenarios must put to the test all included features to validate our software efficiency. We proposed in [23] a preliminary set of typical dangerous contexts to integrate in our safety solution. Nevertheless, we chose to reduce the integrated contexts to the adequate set which allows to encompass all software features. Consequently, one context has been set aside compared to [23]: the time limitation for oven door opening is not included. The *time management* functionality, which is tested here, is actually already contained in other scenarios. Thus, the effective risks coverage for the INOVUS prototype is synthesized in Table 3.2.

Five ADL scenarios have been established along with a representation of the Domus smart apartment. The test environment embeds electromagnetic, infrared, smoke, and temperature sensors, distributed within six areas. These resources lead to a light home monitoring with user localization, tracking for stove elements use, inspection

3.6. VALIDATION RESULTS

for critical environmental factors level, and temporal data. This environment also offers screens, speakers, and lights in each room for user interaction.

Table 3.2: Effective risk coverage in the INOVUS prototype.

Risks	Contexts
Burns	Hot surfaces (oven front or hotplate)
Fires	Food forgotten while heating
	No surveillance in home
	People leaving home while food heating
Intoxication	Smoke detection in the kitchen

Contexts in Table 3.2 were defined with detection, information, and reaction strategies using various INOVUS bundles. As a result, several running sequences in the ADL simulator for each test scenario lead to the successful detection and management for each risk situation. This confirms that our approach is useful. Once it is done, it can be experimented in real settings.

These outcomes may be discussed as they are based on simulated input, however similar tests in a real environment should reach analogous results since the INOVUS software is designed to be fully configurable and unrelated from the targeted system.

XML semantics for data description have shown some significant drawbacks. Risk contexts to monitor must be defined by-hand, following the chosen environment description. This process is highly controversial since the slightest configuration error generates system failure.

Moreover, human-machine interactions could be enhanced with graphical interfaces for software configuration instead of the in-line XML configuration. This should offer user-friendly configuration features, especially for environment and risk configurations definition.

Finally, the INOVUS project scope can be discussed. Indeed, the covered risks collection may appear very light here. However, the integrated risk coverage mainly depends on the available technologies in the targeted environment: the more distributed technologies in the home you have, the more environmental factors you can sense, and the largest risks coverage you can reach. The INOVUS project focuses on safety issues in cooking-related contexts, nevertheless it can be targeted at various risks sources (e.g. wet floors, falls detection). According to the accessible sensors, the INOVUS project could be deployed within the entire home, hence monitoring more activities of daily living. The purpose of the INOVUS project would then be extended to home safety in general.

3.7 Conclusion

This paper presented the INOVUS system and its associated development approach based on formal specifications, leading to a functional prototype. Main elements on this projects have been exposed, including model checking and OSGi-based development in programming. Using such techniques, the INOVUS project aims to tackle identified issues in existing home safety technologies for stove-related hazards.

Several elements have to be taken in account for future improvements on this project. First, the embedded artificial intelligence should be improved with self-learning features, hence offering a better user experience with a completely transparent technology. Second, as XML-based data description techniques do not appear very convenient, efforts in this way should be conducted, for instance using ontology-based solutions [2] and graphical configuration interfaces. Finally, full benefits from model checking integration in our software design motivates the use of formal specifications in the development of critical systems. Work should thus be conducted to explore these techniques in smart-home-related fields.

– *FIN DE L'ARTICLE* –

3.8 Compléments sur le prototype Java

L'article présenté dans ce chapitre a permis d'expliquer la phase de développement du prototype obtenu lors de ces travaux de maîtrise. La section 3.4 a notamment détaillé l'utilisation du cadriciel OSGi et de l'architecture de composants qui en résulte. La section 3.5 a d'autre part introduit la méthodologie de validation suivie. Cette validation du prototype amène, dans le même temps, la validation de l'approche de développement choisie. Cependant, certains aspects de ces éléments méritent d'être explicités un peu plus, tels que les ressources nécessaires au fonctionnement de l'application, ou encore une illustration concrète du prototype en exécution. C'est là l'objectif de cette section.

3.8.1 Données d'entrée de l'application

Les sections 3.4 et 3.5 ont indiqué les données d'entrée à fournir, aussi bien à l'application INOVUS qu'au simulateur d'AVQ. Afin d'illustrer la nature de ces informations, l'élaboration de ces fichiers d'entrée est explicitée ici.

Représentation de l'environnement cible

Le but de l'application INOVUS étant de monitorer un environnement matériel, il est nécessaire d'en connaître la composition. Il faut donc dresser une nomenclature des différents éléments constituant l'espace concerné, aussi bien en terme d'organisation spatiale que d'équipement technologique.

Dans le cas de l'appartement-test du laboratoire Domus, la nomenclature utilisée est la suivante :

- les éléments de cuisine présents (i.e. four et plaques chauffantes) sont décrits par un nom, un identifiant, et un type.
- les pièces de vie (nom, identifiant), comportant chacune :
 - la liste des capteurs disponibles dans la pièce. Chaque capteur est défini par un nom et un identifiant, et est accompagné de la liste de ses états possibles. Si le capteur est lié à un élément de cuisson, on précise de quel élément il s'agit.

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

- la liste des actuateurs disponibles dans la pièce. Chaque actuateur est défini par un nom et un identifiant.

Les attributs accompagnant chacun des éléments ci-dessus ne sont pas spécifiques à l'exemple de l'habitat-test, mais sont à fournir quelque soit l'environnement ciblé, afin d'identifier et de lier correctement les éléments entre eux.

Cette description de l'environnement peut ensuite être retranscrite dans un fichier XML selon une grammaire propre à l'application INOVUS, apportant de cette manière à notre application la connaissance de l'environnement sur lequel elle est déployée. On obtient ainsi un fichier de description de l'environnement d'exécution interprétable par le logiciel. Un extrait du fichier XML utilisé pour INOVUS est donné en B.2.

Représentation d'une situation dangereuse

Les précédentes sections ont évoqué à plusieurs reprises la nécessité de définir les contextes de risques que l'on souhaite intégrer à l'application. C'est en se basant sur ces ressources que l'application est en mesure de détecter chacun des risques couverts, et de fournir une rétroaction adaptée.

Pour détecter efficacement une situation dangereuse, il est donc nécessaire d'établir précisément les éléments caractérisant chacun des contextes de risque (i.e. les facteurs à mesurer, les capteurs concernés, ainsi que leurs valeurs critiques).

D'autre part, pour chacun des contextes définis, il faut préciser les règles de diffusion d'informations et de réactions à appliquer. Ces règles permettent d'indiquer d'une part les actuateurs à utiliser et les informations à fournir par leur biais, d'autre part le type de réaction à initier (i.e. contacter un aidant, un service d'urgence, ou arrêter un élément de cuisson).

À titre d'exemple, l'un des contextes de risques intégré dans ce prototype concerne la détection d'une surface chaude (cf. 3.6), défini de la manière suivante :

Contexte *Surface chaude* Lorsqu'un élément de la cuisinière est utilisé, la température de la zone de cuisson concernée augmente de manière importante, présentant de ce fait un risque imminent de brûlure (par contact) pour un utilisateur se trouvant dans l'entourage direct de la cuisinière. Si le four est en fonction, la surface présentant un danger est alors la porte du

3.8. COMPLÉMENTS SUR LE PROTOTYPE JAVA

four. Il est donc nécessaire d'informer l'utilisateur de la présence d'un risque de brûlure lorsqu'il se trouve à proximité de la cuisinière en fonctionnement, tout en lui précisant l'élément de cuisson concerné:

Compte-tenu de la définition du risque fournie ci-dessus, l'ensemble des éléments à regrouper pour que ce risque se concrétise sont les suivants :

- Un élément de cuisson doit être en fonction.
- L'utilisateur doit se trouver à proximité de la cuisinière.

En se basant sur les capteurs présents dans notre environnement d'étude, les contraintes sur les capteurs environnementaux sont les suivantes :

- Un élément de cuisson doit avoir son capteur d'utilisation à l'état *ON*.
- Le capteur de position de la cuisine doit être à l'état *ON*.

Ces deux éléments constituent notre stratégie de détection du contexte *Surface chaude* : dès lors que les capteurs concernés se trouvent tous deux dans les valeurs critiques, alors on considère que le danger est présent.

On raisonne de la même manière pour définir la stratégie d'information, en précisant pour chaque risque les acteurs concernés par la diffusion d'information, la fréquence de rappel de l'information, ainsi que le message à envoyer.

En ce qui concerne la stratégie de réaction, on va aller indiquer les éléments dont il faut prendre le contrôle ainsi que la commande à envoyer. Dans le cas où l'on viendrait utiliser le service d'information d'une personne extérieure (e.g. aidant ou service d'urgence), on préciserait le message à envoyer ainsi que le destinataire du message. Notons toutefois que ces fonctionnalités n'ont été que partiellement implantées puisqu'elles sont très dépendantes de l'environnement étudié. Dans notre cas, l'utilisation du simulateur ne permet pas leur mise en oeuvre. Seule la prise de contrôle d'un élément de cuisson a donc été intégrée.

Ces stratégies peuvent ensuite être transposées dans un fichier XML, toujours en suivant la grammaire définie pour l'application. Les fichiers XML correspondant à l'exemple présenté ci-dessus sont disponibles à l'annexe B.2.

Cet exemple très simple montre la nature de tâches à effectuer pour pouvoir

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

intégrer un risque au sein de l'application INOVUS. Ces tâches doivent être réitérées pour chaque nouveau contexte à prendre en compte, sachant que certains contextes peuvent regrouper une combinaison bien plus complexe de facteurs, en fonction des capteurs à disposition dans l'environnement et du risque que l'on souhaite modéliser. Dans certains cas, il peut être nécessaire de prendre en compte non seulement l'état d'un capteur à un instant t , mais aussi aux instants $t-1$ et $t+1$ par exemple. Ce genre de contraintes viennent augmenter considérablement la complexité d'élaboration des stratégies de détection.

3.8.2 Illustration du processus de gestion des risques

Afin d'illustrer le processus de gestion des risques mis en oeuvre par INOVUS, nous allons présenter ici un exemple de gestion du contexte *Surface chaude*.

Simulation du scénario

On se place dans un scénario où une personne seule chez elle réalise les tâches suivantes :

- Lire dans la salle à manger.
- Préparer une soupe à partir d'une préparation instantanée.
- Manger la soupe dans la salle à manger.

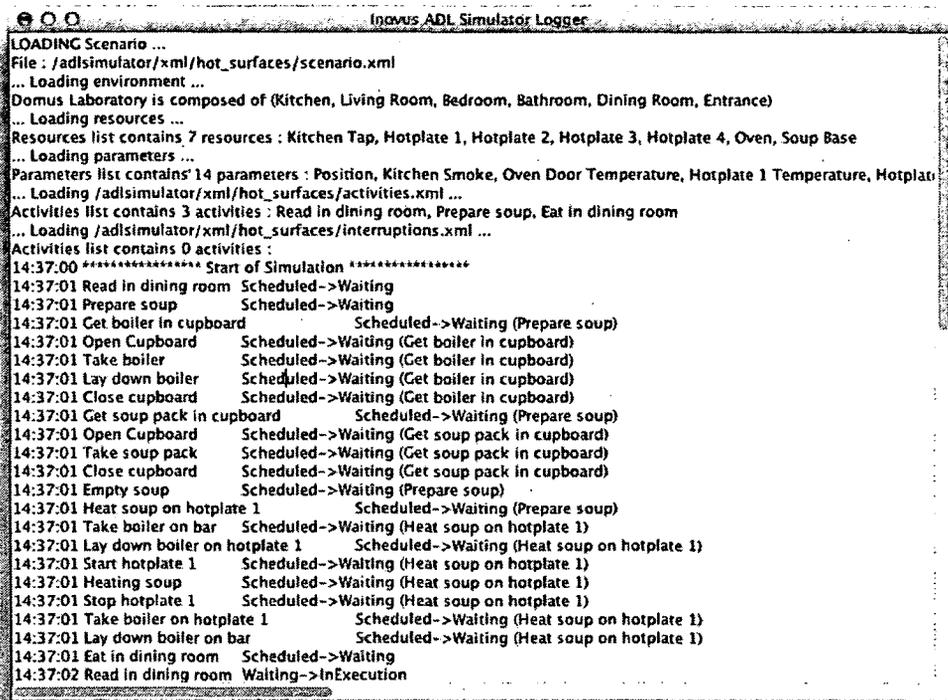
La figure 3.7 illustre le chargement de ce scénario, ainsi que l'exécution des tâches qui le composent. Cette étape est réalisée par le simulateur d'AVQ qui agit alors comme étant le système cible (i.e. la couche supérieure dans l'architecture présentée à la figure 3.3).

Le simulateur exécute ainsi le scénario, au cours duquel le contexte *Surface chaude* apparaît puisque l'utilisateur se retrouve à faire chauffer sa soupe sur l'une des plaques chauffantes, tout en étant à proximité.

Surveillance des activités

Nous avons présenté à la figure 3.5 l'algorithme de surveillance suivi par INOVUS. L'application recueille en continu l'état courant du système cible et l'analyse pour identifier les potentiels dangers présents. Un *logger* a été développé afin d'avoir un

3.8. COMPLÉMENTS SUR LE PROTOTYPE JAVA



```
LOADING Scenario ...
File : /adlsimulator/xml/hot_surfaces/scenario.xml
... Loading environment ...
Domus Laboratory is composed of (Kitchen, Living Room, Bedroom, Bathroom, Dining Room, Entrance)
... Loading resources ...
Resources list contains 7 resources : Kitchen Tap, Hotplate 1, Hotplate 2, Hotplate 3, Hotplate 4, Oven, Soup Base
... Loading parameters ...
Parameters list contains 14 parameters : Position, Kitchen Smoke, Oven Door Temperature, Hotplate 1 Temperature, Hotplate 2 Temperature, Hotplate 3 Temperature, Hotplate 4 Temperature, Oven Temperature, Kitchen Tap Temperature, Kitchen Tap Position, Hotplate 1 Position, Hotplate 2 Position, Hotplate 3 Position, Hotplate 4 Position
... Loading /adlsimulator/xml/hot_surfaces/activities.xml ...
Activities list contains 3 activities : Read in dining room, Prepare soup, Eat in dining room
... Loading /adlsimulator/xml/hot_surfaces/interruptions.xml ...
Activities list contains 0 activities :
14:37:00 ***** Start of Simulation *****
14:37:01 Read in dining room Scheduled->Waiting
14:37:01 Prepare soup Scheduled->Waiting
14:37:01 Get boiler in cupboard Scheduled->Waiting (Prepare soup)
14:37:01 Open Cupboard Scheduled->Waiting (Get boiler in cupboard)
14:37:01 Take boiler Scheduled->Waiting (Get boiler in cupboard)
14:37:01 Lay down boiler Scheduled->Waiting (Get boiler in cupboard)
14:37:01 Close cupboard Scheduled->Waiting (Get boiler in cupboard)
14:37:01 Get soup pack in cupboard Scheduled->Waiting (Prepare soup)
14:37:01 Open Cupboard Scheduled->Waiting (Get soup pack in cupboard)
14:37:01 Take soup pack Scheduled->Waiting (Get soup pack in cupboard)
14:37:01 Close cupboard Scheduled->Waiting (Get soup pack in cupboard)
14:37:01 Empty soup Scheduled->Waiting (Prepare soup)
14:37:01 Heat soup on hotplate 1 Scheduled->Waiting (Prepare soup)
14:37:01 Take boiler on bar Scheduled->Waiting (Heat soup on hotplate 1)
14:37:01 Lay down boiler on hotplate 1 Scheduled->Waiting (Heat soup on hotplate 1)
14:37:01 Start hotplate 1 Scheduled->Waiting (Heat soup on hotplate 1)
14:37:01 Heating soup Scheduled->Waiting (Heat soup on hotplate 1)
14:37:01 Stop hotplate 1 Scheduled->Waiting (Heat soup on hotplate 1)
14:37:01 Take boiler on hotplate 1 Scheduled->Waiting (Heat soup on hotplate 1)
14:37:01 Lay down boiler on bar Scheduled->Waiting (Heat soup on hotplate 1)
14:37:01 Eat in dining room Scheduled->Waiting
14:37:02 Read in dining room Waiting->InExecution
```

Figure 3.7 – Extrait du logger du simulateur d’AVQ.

suivi visuel des dangers détectés lorsqu’il y en a. Dans notre exemple, la surface chaude de la plaque de cuisson utilisée a bien été détectée, comme le montre la figure 3.8.

Gestion de risque

À chaque analyse de l’environnement, l’application retourne la liste des situations dangereuses trouvées. On remarque sur la figure 3.8 que le risque de brûlure est bien présent, sur l’ensemble des cinq vérifications consécutives visibles. L’application précise de plus l’identifiant de l’élément de cuisson concerné, ainsi que l’heure de détection. Ces données ne sont pas directement fournies à l’utilisateur, mais récupérées lors de la mise en oeuvre des stratégies d’information et de réaction.

Pour le risque *Surface chaude*, la stratégie d’information implique la diffusion d’un message d’alerte à destination de l’usager, et cela suivant une fréquence de rappel fournie au sein de la stratégie, avec le message à diffuser.

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

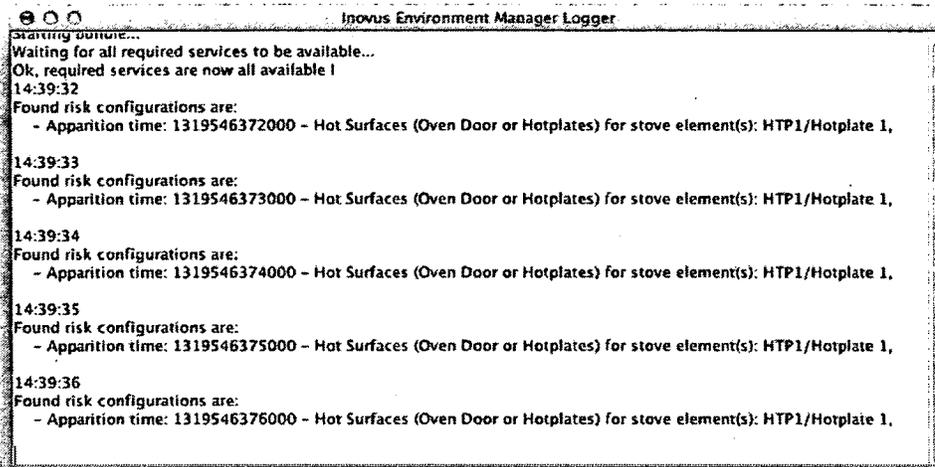


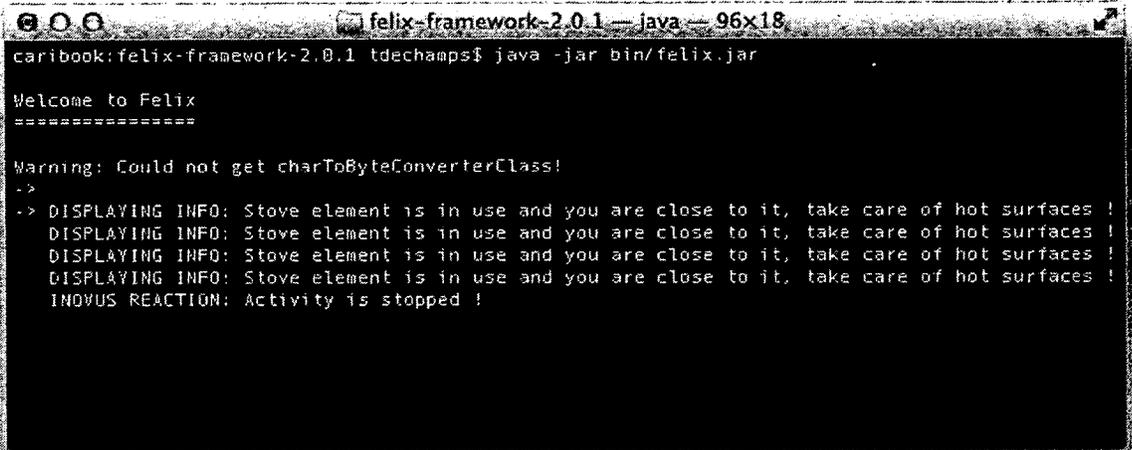
Figure 3.8 – Extrait du logger des dangers détectés par l'application INOVUS.

D'autre part, la gestion d'un risque est démarrée dès sa détection en suivant un séquençement similaire à la figure 3.6. De cette manière, la mise en application des stratégies d'information et de réaction pour le contexte *Surface chaude* de notre exemple est automatiquement initiée. La figure 3.9 présente les interactions qui en découlent.

On remarque ainsi que l'avertissement concernant la surface chaude a bien été envoyé, et cela suivant la fréquence de rappel prédéfinie. D'autre part, bien que ce danger ne nécessite pas nécessairement une prise de contrôle de l'élément de cuisson utilisé, une commande de la plaque chauffante a tout de même été intégrée à la stratégie de réaction afin de montrer que l'intervention dans le système cible est bien effective.

Ainsi, sur la figure 3.9, les lignes débutant par `DISPLAYING INFO` correspondent au partage d'information avec l'utilisateur, et sont générées par le bundle *Information Provider*. La ligne débutant par `INOVUS REACTION` correspond quant à elle à la simulation d'une prise de contrôle dans l'environnement cible, et est donc renvoyée au simulateur par le biais du bundle *Stove Controller* et de la couche d'adaptation.

3.8. COMPLÉMENTS SUR LE PROTOTYPE JAVA



```
caribook:felix-framework-2.0.1 tdechamps$ java -jar bin/felix.jar
Welcome to Felix
=====
Warning: Could not get charToByteConverterClass!
->
-> DISPLAYING INFO: Stove element is in use and you are close to it, take care of hot surfaces !
DISPLAYING INFO: Stove element is in use and you are close to it, take care of hot surfaces !
DISPLAYING INFO: Stove element is in use and you are close to it, take care of hot surfaces !
DISPLAYING INFO: Stove element is in use and you are close to it, take care of hot surfaces !
INOVUS REACTION: Activity is stopped !
```

Figure 3.9 – Extrait du logger de rétroactions de l'application INOVUS.

L'exemple présenté dans cette section aura permis d'illustrer concrètement deux piliers fondamentaux de ces travaux. D'une part, les clarifications apportées sur la phase de configuration de l'application INOVUS permettent de mieux cibler les éléments qui lui sont nécessaires pour monitorer un environnement intelligent. D'autre part, le fonctionnement détaillé de l'application aura été explicité à l'aide de l'article scientifique, puis par l'exemple ci-dessus.

C'est donc ce qui conclut ce chapitre sur le développement du prototype d'INOVUS.

CHAPITRE 3. VALIDATION D'APPROCHE À L'AIDE D'UN PROTOTYPE

Conclusion

Ce mémoire a fait état des travaux conduits dans le cadre de ce programme de maîtrise. Il a ainsi présenté les différentes phases du développement d'un premier prototype fonctionnel d'application, destiné à effectuer une preuve de concept concernant la couche logicielle du système INOVUS. Pour parvenir à ce prototype, l'objectif de ces travaux de maîtrise était de venir proposer une approche de développement à base de spécifications formelles du système, et plus spécifiquement de la vérification de modèles. Les résultats ainsi obtenus ont permis de valider l'approche proposée.

Les objectifs des travaux de ce mémoire ont été dans la majorité atteints avec succès. D'une part, la revue de littérature du chapitre 1 a permis d'identifier correctement les risques à domicile, et d'en établir une hiérarchie. D'autre part, l'étude des solutions technologiques existantes a fait ressortir un panel intéressant de produits, chacun présentant leurs points forts, ainsi que leurs faiblesses. Ces éléments ont pu ensuite être intégrés parmi les critères indispensables à couvrir pour le prototype à mettre au point, fournissant ainsi une série de contraintes fortes pour la conception et le développement de l'application.

En ce qui concerne la phase de spécifications formelles du projet, les résultats présentés au chapitre 2 ont fait ressortir que l'utilisation du vérificateur de modèle a été réellement bénéfique pour la conception et le développement de l'application. Cependant, une étude approfondie serait nécessaire afin de vérifier si ALLOY constitue bien l'outil le plus approprié dans le cadre du projet INOVUS. L'objectif de cette phase était d'évaluer l'intégration d'un tel outil dans le développement d'une technologie pervasive, et les résultats obtenus suite à cette utilisation apparaissent satisfaisants.

Enfin, le prototype Java produit a fait naître des perspectives plus qu'encoura-

CONCLUSION

geantes pour l'élaboration des versions ultérieures de l'application. Les risques intégrés dans ce premier incrément sont détectés avec succès, et les réactions qui y sont associées sont gérées correctement. Cela nous indique donc que la solution retenue permet bien de rencontrer nos exigences en terme de sécurité de la personne lors de l'utilisation de la cuisinière.

En se basant sur ces éléments, la preuve de concept recherchée concernant INOVUS est clairement établie, et les résultats positifs concernant le prototype produit permette de valider la proposition d'approche de développement à base de méthodes formelles.

Dans la mesure où l'application présentée dans ce mémoire n'est qu'à un stade embryonnaire, les axes d'améliorations possibles sont variés. Cependant, quelques aspects paraissent assez fondamentaux pour contribuer efficacement à l'évolution de ce prototype.

Tout d'abord, l'engin de raisonnement embarqué dans l'application pourrait être largement amélioré, apportant ainsi plusieurs éléments intéressants : il serait alors envisageable d'intégrer des techniques d'intelligence artificielle permettant de l'auto-apprentissage, ce qui permettrait d'optimiser très largement la phase de configuration de l'application en réduisant très nettement la quantité d'informations à fournir au système pour le rendre opérationnel.

Ensuite, le système de description des données de l'application doit être amélioré. En effet, l'approche adoptée dans ces travaux de maîtrise a permis de fournir une représentation, facilement interprétable par l'application, de l'environnement que l'on veut surveiller. Cela s'est fait en adoptant des règles assez primitives de relations entre les différentes entités de notre environnement, et en le retranscrivant à l'aide du langage de description XML directement dans l'application. Une telle technique est envisageable pour un environnement de petite envergure, néanmoins les limites de ce choix serait très rapidement atteintes dans le cas d'un système plus important. Il semblerait donc pertinent de venir renforcer ce point en adoptant des outils plus robustes de description de données, telles que les ontologies.

Par ailleurs, de par son statut de projet de recherche universitaire, INOVUS bénéficie d'une certaine flexibilité quant aux contraintes de conception et de développement.

CONCLUSION

Cependant, il est à noter que le développement d'appareils électroménagers destinés au grand public doit se faire en respectant des normes et certifications strictes, toujours dans un souci (entre autres) de sécurité de l'utilisateur. De ce fait, dans l'hypothèse d'une commercialisation de cette solution technologique, ces normes en impacteraient indéniablement le développement. Ces modifications interviendraient alors aussi bien en termes de conception, d'architecture, que de développement. Si le futur du projet INOVUS s'oriente dans cette direction, alors il serait intéressant de prendre en compte ces considérations dès les prochaines étapes du projet.

Enfin, au vu des bénéfices apportés par la phase de spécifications formelles du système et des limites du vérificateur de modèle utilisé, il serait intéressant d'explorer plus en détails les autres techniques de spécifications formelles existantes afin de trouver l'outil le mieux adapté. Cette remarque s'applique d'ailleurs non seulement au projet INOVUS, mais aussi à bien d'autres projets développés dans un habitat intelligent. En effet, la majorité des technologies contribuant au maintien à domicile présentent des similitudes avec les systèmes critiques habituels, et l'utilisation de spécifications formelles pourrait dès lors leur être tout aussi bénéfique.

Maintenir efficacement à domicile des personnes déficientes implique automatiquement l'assurance de la sécurité de la personne. Les inconvénients qui ressortent des solutions actuelles pour la sécurité à domicile, couplés aux résultats des nombreuses études sur les accidents domestiques, prouvent la nécessité de mener des travaux sur ce secteur. En se basant sur l'ensemble des éléments présentés dans ce mémoire, le projet INOVUS semble apporter une solution viable pour assurer la sécurité d'un usager lors de l'utilisation de sa cuisinière. Ce constat est établi en regard des lacunes identifiées parmi les solutions technologiques existantes.

L'utilisation d'un habitat intelligent pour contribuer au maintien à domicile des personnes âgées ou souffrant de troubles cognitifs se place comme l'une des solutions émergentes en fort développement au cours de ces dernières années.

Dans cette même voie, le résultat final de ces travaux de maîtrise illustre les prémices d'une solution de sécurité qui se montre plus qu'encourageante pour maintenir efficacement une personne seule à domicile, tout particulièrement dans le cas de personnes âgées ou souffrant de troubles cognitifs. Les résultats fournis par ce premier prototype

CONCLUSION

ne sont qu'un aperçu du potentiel que présentent les habitats intelligents en terme de sécurité de la personne, il semble donc indispensable de continuer les travaux dans cette voie.

Annexe A

Détails sur l'utilisation du vérificateur de modèle Alloy

Cette annexe présente des extraits du modèle formel développé en ALLOY dans le cadre du projet INOVUS. Les principaux éléments en terme de structure et comportement du modèle sont expliqués ici.

Afin d'illustrer le processus d'évaluation de propriétés conduit par *Alloy Analyzer*, cette annexe fournit d'autre part un exemple d'exigence, traduite avec ALLOY. La liste complète des propriétés élaborées lors de la vérification de modèle suit cet exemple.

ANNEXE A. DÉTAILS SUR L'UTILISATION DU VÉRIFICATEUR DE MODÈLE ALLOY

A.1. PRÉSENTATION DU MODÈLE DÉVELOPPÉ AVEC ALLOY

A.1 Présentation du modèle développé avec Alloy

A.1.1 Structure du modèle

La construction du modèle formel s'est basée sur un diagramme de classe simplifié du système, repris par la figure A.1. Ce diagramme présente les différentes entités utilisées dans le modèle ALLOY pour représenter le système à développer.

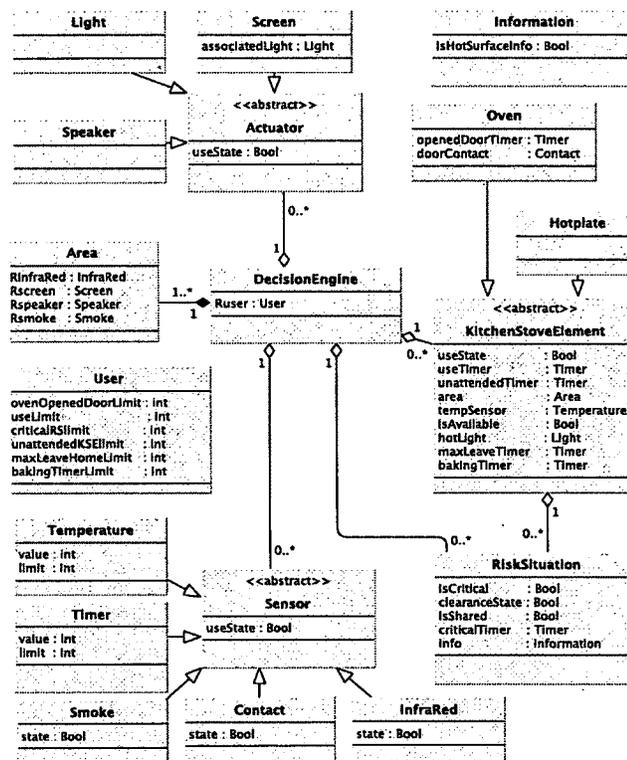


Figure A.1 – Diagramme de classe utilisé pour construire le modèle ALLOY.

Cette représentation a permis de développer le code complet du modèle, aussi bien en terme d'ensembles (signatures), que de relations (fact et cardinalités) et dynamisme (prédicats).

Relations d'ensembles avec Alloy

L'extrait de code présenté ci-dessous correspond à la modélisation de l'engin de raisonnement, qui constitue l'élément clé du système. De ce fait, la signature comporte

ANNEXE A. DÉTAILS SUR L'UTILISATION DU VÉRIFICATEUR DE MODÈLE ALLOY

l'ensemble des attributs nécessaires à la représentation du modèle de solution. Chacun de ces attributs appartient à un ensemble du système (e.g. `Ruser`, `Rsensor`, etc...) et est associé à une cardinalité (one, lone, ou set).

La section `fact` décrit ensuite les règles fixées sur les relations entre les ensembles du modèle. On retrouve donc ici les notions de partitions, disjonctions, ou appartenance.

Programme A.1 – Code ALLOY de l'entité `DecisionEngine`.

```
1  -- Signature
2  sig DecisionEngine
3  {
4    Ruser : one User,
5    Rsensor : set Sensor,
6    Ractuator: set Actuator,
7    RkitchenStoveElement : set KitchenStoveElement,
8    RriskSituation : set RiskSituation,
9    Rarea : set Area,
10  flag : one Int
11  }
12
13  -- Rules concerning relations between sets
14  fact
15  {
16    all d:DecisionEngine {
17      d.RkitchenStoveElement.hotLight in d.Ractuator
18      d.RkitchenStoveElement.tempSensor in d.Rsensor
19      d.RkitchenStoveElement.area in d.Rarea
20      d.RkitchenStoveElement.useTimer in d.Rsensor
21      d.RkitchenStoveElement.unattendedTimer in d.Rsensor
22      d.RkitchenStoveElement.maxLeaveTimer in d.Rsensor
23      d.RkitchenStoveElement.bakingTimer in d.Rsensor
24      d.RkitchenStoveElement.doorContact in d.Rsensor
25      d.RkitchenStoveElement.openedDoorTimer in d.Rsensor
26      d.RkitchenStoveElement.rs = d.RriskSituation
27      d.Rarea.Rscreen.associatedLight in d.Ractuator
28      d.Rarea.RinfraRed in d.Rsensor
29      d.Rarea.Rspeaker in d.Ractuator
30      d.Rarea.Rscreen in d.Ractuator
31      d.Rarea.Rsmoke in d.Rsensor
32      d.RriskSituation.criticalTimer in d.Rsensor
33      #d.Ruser = 1
34    }
35  }
36
37  -- Sets partitions
38  fact
```

A.1. PRÉSENTATION DU MODÈLE DÉVELOPPÉ AVEC ALLOY

```
39 {
40   Partition2[KitchenStoveElement.hotLight, Screen.associatedLight, Light]
41   all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.hotLight | Injective[k->t]
42   all d:DecisionEngine, s:Screen, a:s.associatedLight |
43     s in d.Ractuator and Injective[s->a]
44
45   KitchenStoveElement = Oven
46   =>
47   {
48     Partition6[KitchenStoveElement.useTimer, KitchenStoveElement.unattendedTimer,
49       KitchenStoveElement.openedDoorTimer, KitchenStoveElement.maxLeaveTimer,
50       KitchenStoveElement.bakingTimer, RiskSituation.criticalTimer, Timer]
51     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.unattendedTimer | Injective[k->t]
52     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.useTimer | Injective[k->t]
53     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.openedDoorTimer | Injective[k->t]
54     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.maxLeaveTimer | Injective[k->t]
55     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.bakingTimer | Injective[k->t]
56     Injective[criticalTimer]
57   } else
58   {
59     Partition5[KitchenStoveElement.useTimer, KitchenStoveElement.unattendedTimer,
60       KitchenStoveElement.maxLeaveTimer, KitchenStoveElement.bakingTimer,
61       RiskSituation.criticalTimer, Timer]
62     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.unattendedTimer | Injective[k->t]
63     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.useTimer | Injective[k->t]
64     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.maxLeaveTimer | Injective[k->t]
65     all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.bakingTimer | Injective[k->t]
66     Injective[criticalTimer]
67   }
68
69   all d:DecisionEngine, k:d.RkitchenStoveElement, t:k.tempSensor | Injective[k->t]
70
71   all d:DecisionEngine, a, a':d.Rarea, i:InfraRed | a!=a' and i in a.RinfraRed
72   => i !in a'.RinfraRed
73
74   all d:DecisionEngine, a, a':d.Rarea, s:Speaker | a!=a' and s in a.Rspeaker
75   => s !in a'.Rspeaker
76
77   all d:DecisionEngine, a, a':d.Rarea, s:Screen | a!=a' and s in a.Rscreen
78   => s !in a'.Rscreen
79
80   all d:DecisionEngine, disj a, a':d.Rarea | isTrue[a.RinfraRed.state]
81   => isFalse[a'.RinfraRed.state]
82 }
```

ANNEXE A. DÉTAILS SUR L'UTILISATION DU VÉRIFICATEUR DE MODÈLE ALLOY

Héritage avec Alloy

Parmi les fonctionnalités intéressantes d'ALLOY dans la construction de notre modèle, l'héritage permet de simplifier grandement le raisonnement sur les ensembles. On peut ainsi définir des classes mères/filles. Le programme A.2 décrit par exemple la transposition en ALLOY des classes des éléments de cuissons présentées à la figure A.1.

Programme A.2 – Signature abstraite en ALLOY modélisant un élément de cuisson, accompagnées des signatures dérivées.

```
1 abstract sig KitchenStoveElement
2 {
3   useState : one Bool,
4   useTimer : one Timer,
5   unattendedTimer : one Timer,
6   area : one Area,
7   tempSensor : one Temperature,
8   isAvailable : one Bool,
9   hotLight : one Light,
10  maxLeaveTimer : one Timer,
11  bakingTimer : lone Timer,
12  rs : set RiskSituation
13 }
14
15 sig Hotplate extends KitchenStoveElement
16 {
17 }
18
19 sig Oven extends KitchenStoveElement
20 {
21   openedDoorTimer : Timer,
22   doorContact : Contact
23 }
```

A.1. PRÉSENTATION DU MODÈLE DÉVELOPPÉ AVEC ALLOY

A.1.2 Comportement du modèle

Le programme A.3 présente l'un des prédicats définis pour construire le modèle formel. L'extrait présenté ici modélise le passage pour un capteur de température de l'état froid à l'état défini comme chaud (température mesurée supérieure à une température seuil). Ainsi, le changement d'état du capteur implique pour l'élément de cuisson concerné que la valeur de son capteur est modifiée, et que l'indicateur lumineux de surface chaude associé doit être allumé. Les autres attributs et ensembles sont quant à eux inchangés.

Les prédicats permettent de représenter les aspects dynamiques du système (événements possibles, actions de l'utilisateur, modifications de l'environnement). De tels prédicats ont été définis pour représenter les événements suivants :

- requêtes de l'utilisateur (e.g. mise en route et arrêt d'un élément, configuration de timers, acquittement d'une situation dangereuse) ;
- changements environnementaux (e.g. déplacement de l'utilisateur, changements d'états de capteurs ou d'un élément de cuisson) ;
- détection d'un danger ;
- réaction du système suite à l'apparition d'un danger.

D'une manière générale, la structure d'un prédicat est la suivante : pré-conditions, actions, post-conditions. Les pré-conditions fixe le contexte de réalisation d'un prédicat, alors que les actions correspondent aux modifications à apporter aux ensembles concernés pour représenter un événement précis. Enfin, puisqu'un prédicat modélise le passage du système d'un état à un autre, les post-conditions sont destinées à indiquer que les ensembles non modifiés par le prédicat doivent rester identiques entre l'état de départ et l'état d'arrivée (appels des différents prédicats `NoChange...` définis dans un fichier externe).

Programme A.3 – Prédicat ALLOY modélisant le passage pour un capteur de température de l'état froid à l'état défini comme chaud.

```
1 pred ThresholdTemperatureExceededHot_CAN[d:DecisionEngine,k:d.RkitchenStoveElement]
2 {
3   k.tempSensor.value < k.tempSensor.limit
4 }
5
```

ANNEXE A. DÉTAILS SUR L'UTILISATION DU VÉRIFICATEUR DE MODÈLE ALLOY

```
6
7 pred ThresholdTemperatureExceededHot [d,d' :DecisionEngine,k:d.RkitchenStoveElement]
8 {
9   --Preconditions
10  ThresholdTemperatureExceededHot_CAN[d,k]
11
12  --Actions
13  some risk:RiskSituation
14  {
15    isFalse[risk.isCritical]
16    isFalse[risk.clearanceState]
17    isTrue[risk.info.isHotSurfaceInformation]
18    isTrue[risk.isShared]
19
20    some k':KitchenStoveElement
21    {
22      k in Oven => k' in Oven else k' in Hotplate
23      k'.isAvailable = k.isAvailable
24      k'.useState = k.useState
25      k'.unattendedTimer = k.unattendedTimer
26      k'.area = k.area
27      k'.tempSensor.limit = k.tempSensor.limit
28      k'.tempSensor.value >= k.tempSensor.limit
29      k'.maxLeaveTimer = k.maxLeaveTimer
30      k' in Oven and k in Oven => (k'.openedDoorTimer = k.openedDoorTimer and k'.
        doorContact = k.doorContact)
31      some k'.bakingTimer and some k.bakingTimer => (k'.bakingTimer = k.bakingTimer)
32
33      isTrue[k'.hotLight.useState]
34
35      k'.rs = k.rs + risk
36
37      d'.Ractuator = d.Ractuator - k.hotLight + k'.hotLight
38      d'.RkitchenStoveElement = d.RkitchenStoveElement - k + k'
39      d'.Rsensator = d.Rsensator - k.tempSensor + k'.tempSensor
40    }
41    d'.RiskSituation = d.RriskSituation + risk
42  }
43
44  --Postconditions
45  NoChangeRarea[d,d']
46  NoChangeRuser[d,d']
47 }
```

A.1. PRÉSENTATION DU MODÈLE DÉVELOPPÉ AVEC ALLOY

A.1.3 Exécution de la vérification de modèle

La configuration et le lancement de la vérification sont gérés dans un fichier distinct qui doit comporter les éléments suivants : inclusions de fichiers externes, un *fact* définissant les séquences d'événements possibles, un prédicat d'initialisation des ensembles logiques du modèle, et enfin une instruction *run* qui lance l'exécution selon les options définies (*scope*). Le programme A.4 présente le squelette du fichier d'exécution utilisé pour le prototype d'INOVUS.

Programme A.4 – Squelette du fichier d'exécution utilisé.

```
1  -- Inclusions
2  open ...
3  open ...
4  open util/ordering[DE/DecisionEngine] as SequenceDE
5
6  -- Sequencement
7  fact {
8    init[SequenceDE/first[]]
9    all d:DecisionEngine - SequenceDE/last[] |
10   let d'=SequenceDE/next[d] |
11     some k:d.RkitchenStoveElement, risk:k.rs, newLimit:Int, a1,a2:d.Rarea, s:Smoke |
12     ... liste des predicats executables...
13 }
14
15 -- Initialisation
16 pred init[d:DecisionEngine]
17 {
18   ... initialisation des ensembles dans un etat initial possible...
19 }
20
21 -- Run
22 run {}
23 for 6 int, 3 Area, 60 Sensor, 20 Bool, 1 User, 14 Actuator, 3 RiskSituation, 8
    Information, 1 Oven, 6 Hotplate, 2 DecisionEngine
```

A.2 Liste des propriétés Alloy

A.2.1 Exemple de propriété

Le programme A.5 propose un exemple de propriété établie pour le modèle formel. Il s'agit de la traduction en ALLOY de l'exigence de détection d'une surface chaude en tant que situation à risque.

L'assertion proposée impose donc l'implication suivante :

Si le prédicat modélisant le passage à l'état chaud d'un capteur de température d'une surface est réalisé entre deux états consécutifs n et $n+1$,

Alors l'ensemble des situations dangereuses du système en $n+1$ doit contenir une nouvelle situation dangereuse, qui n'était pas présente à l'état n .

Programme A.5 – Exemple de propriété ALLOY vérifiée par le modèle formel avec *Alloy Analyzer*.

```

1  --15. Exceeding the hot threshold temperature for a temperature
      sensor must be detected as a risk situation.
2
3  assert Prop15
4  (
5    all d,d':DecisionEngine, k:d.RkitchenStoveElement | d'=SequenceDE/next[d]
6      and Env_eThresholdTemperatureExceededHot[d,d',k]
7    =>
8      some rs:d.RiskSituation | rs !in d.RiskSituation and rs in d'.RiskSituation
9      and isFalse[rs.isCritical]
10 )
11 check Prop15 for 5

```

Le mot-clé `check` initie l'évaluation de la propriété par *Alloy Analyzer* sur une instance du modèle, et indique au terme de cette évaluation si des contre-exemples ont pu être trouvés. Comme décrit à la figure 2.3, si aucun contre-exemple n'est retourné, alors la propriété est considérée comme vérifiée.

A.2. LISTE DES PROPRIÉTÉS ALLOY

A.2.2 Liste complète des propriétés établies

La liste ci-dessous indique l'ensemble des propriétés qui ont été établies pour ce prototype du projet INOVUS¹. La rédaction de propriétés n'étant pas une chose aisée, il est arrivé à deux reprises (propriétés 11 et 12) de devoir traduire l'expression littérale de la propriété en 2 assertions, selon le type de risque détecté (i.e. soit surface chaude, soit un autre risque). Un total de 23 assertions, correspondant à 21 propriétés, ont donc été définies pour ce modèle. Notons que ces propriétés concernent essentiellement la détection des situations dangereuses, ainsi que leur traitement.

1. A KitchenStoveElement in use must activate the HotSurface Light.
2. The UseTimer of a KitchenStoveElement must be activated only when this KitchenStoveElement is in use.
3. The UnattendedTimer of a KitchenStoveElement must be activated only when nobody is present in the KitchenStoveElement area.
4. The MaxLeaveTimer of a KitchenStoveElement must be activated only when nobody is present in the home.
5. The BakingTimer of a KitchenStoveElement must be activated only when user sets it.
6. A KitchenStoveElement in use must activate the associated Reminder.
7. The OpenedDoorTimer of an Oven must be activated only when the oven door is opened.
8. For all timers except CriticalTimer, a limit is set and exceeding it must be detected as a RiskSituation.
9. For a CriticalTimer, exceeding the timer limit must set the associated RiskSituation as critical.
10. An activated hot surface Light must be detected as a RiskSituation.

1. L'expression littérale d'une propriété cible une fonctionnalité bien précise, et la rédaction se fait donc très rigoureusement. Afin de ne pas perdre de sens au travers d'une traduction éventuelle, les propriétés ont été laissées dans leur langue de rédaction.

ANNEXE A. DÉTAILS SUR L'UTILISATION DU VÉRIFICATEUR DE MODÈLE ALLOY

11. The user must be informed (actuators activation) when a RiskSituation is detected.
12. When a RiskSituation is removed, the associated information (actuators) must be stopped.
13. The user must be informed when a critical RiskSituation is declared.
14. Smoke detection must be declared as a critical RiskSituation.
15. Exceeding the hot threshold temperature for a TempSensor must be detected as a RiskSituation.
16. Exceeding the cold threshold temperature for a TempSensor must remove the associated RiskSituation.
17. A KitchenStoveElement must be stopped and set as not available when a critical RiskSituation is declared.
18. Exceeding the CriticalTimer limit must set the associated RiskSituation as critical and the associated KitchenStoveElement as not available.
19. A non-available KitchenStoveElement cannot be used.
20. The clearance of a RiskSituation must reset the associated Actuators.
21. The clearance of all critical RiskSituations for a KitchenStoveElement must delete the associated information and set the KitchenStoveElement as available.

Annexe B

Détails sur le prototype d'Inovus validant de l'approche de développement

Cette annexe fournit des détails sur le prototype Java développé afin de valider l'approche de développement proposée. Ainsi, les spécificités de développement liées à l'utilisation du cadriciel OSGi sont présentées ici. D'autre part, la seconde section de cette annexe décrit la nature des données d'entrée nécessaires au fonctionnement de l'application INOVUS, ainsi que leur description à l'aide du langage XML.

ANNEXE B. DÉTAILS SUR LE PROTOTYPE D'INOVUS VALIDANT DE L'APPROCHE
DE DÉVELOPPEMENT

B.1. SPÉCIFICITÉS LIÉES À L'UTILISATION D'OSGi

B.1 Spécificités liées à l'utilisation d'OSGi

Le cadriciel OSGi est fourni avec une API proposée et maintenue par l'*OSGi Alliance*¹ [6], permettant de développer des composants et de les exécuter à l'aide de l'une des implémentations existantes du framework (e.g. Eclipse Equinox, Apache Felix, Oscar, Knoplerfish). Cette section présente donc les quelques spécificités liées à l'utilisation d'OSGi dans le cadre du projet INOVUS.

B.1.1 Exécution d'un bundle OSGi

Toute application développée avec OSGi nécessite une implémentation du cadriciel afin de démarrer et gérer les différents composants, aussi bien en termes de services que de cycle de vie. L'outil choisi pour remplir cette fonction est *Apache Felix*² [33]. Il s'agit d'un outil développé par la fondation *Apache*, et qui offre une implémentation simple, intuitive, et légère d'OSGi. La documentation sur l'outil étant d'autre part très complète, ces éléments ont donc motivé le choix d'un tel outil.

Chaque composant développé avec OSGi comporte un fichier *activateur* qui permet de définir le comportement du bundle lorsqu'il est démarré. Ce fichier permet de plus de gérer les services mis à disposition ou consommés par le bundle.

Le programme B.1 montre un exemple de fichier *activateur*, pour le composant *Reaction Manager* chargé de gérer la mise en oeuvre de réactions face à un contexte de risque.

Cet exemple fait ressortir deux éléments fondamentaux au sein de la méthode `start` qui démarre le composant.

D'une part, les spécifications OSGi fournissent des outils automatisés de recherche de service. Dans cet exemple, ce sont les `ServiceTracker` qui sont utilisés puisqu'ils permettent de récupérer automatiquement la référence à un service dès sa mise à disposition. Ainsi, la recherche du service fourni par le *Information Provider* effectuée entre les lignes 6 et 13 du programme B.1 est réitérée pour chacun des services requis par le bundle.

1. OSGi Alliance - <http://www.osgi.org/>

2. Apache Felix - <http://felix.apache.org/>

ANNEXE B. DÉTAILS SUR LE PROTOTYPE D'INOVUS VALIDANT DE L'APPROCHE DE DÉVELOPPEMENT

D'autre part, lorsqu'un composant offre un service, il doit l'enregistrer à son démarrage auprès du cadriciel OSGi. C'est ce qui est effectué à la ligne 19 du programme B.1.

Programme B.1 – Extrait du fichier *Activator.java* du bundle *Reaction Manager*.

```
1 public class Activator implements BundleActivator {
2
3     @Override
4     public void start(BundleContext context) throws Exception {
5
6         ServiceTracker shareInfoServiceTracker = new ServiceTracker(
7             context,
8             context.createFilter("(objectClass="
9                 + IShareInfoService.class.getName()
10                + ")"),
11            null);
12
13        shareInfoServiceTracker.open();
14
15        /* ... Instanciation des ServiceTracker pour les autres services requis ... */
16
17        /* ... Instanciation des autres elements du bundle ... */
18
19        context.registerService(
20            IReactionsLauncherService.class.getName(),
21            new ReactionsLauncherService(receiver),
22            null);
23    }
24
25    @Override
26    public void stop(BundleContext context) throws Exception {
27    }
28 }
```

Il ne s'agit ici que d'un exemple possible de fichier *activateur*. Les spécifications OSGi offrent différentes classes pour gérer efficacement et automatiquement l'enregistrement et la recherche d'un service.

Enfin, la déclaration d'un service dans un composant nécessite aussi de respecter quelques règles. La section suivante fournit donc un exemple de déclaration de service, toujours dans le cas du bundle *Reaction Manager*.

B.1. SPÉCIFICITÉS LIÉES À L'UTILISATION D'OSGI

B.1.2 Définition d'un service

La définition d'un service s'effectue de manière très simple avec OSGi, et se décompose en deux éléments :

- Une interface pour accéder au service.
- Une classe qui implémente cette interface, et qui spécifie le rôle des méthodes du service.

Le programme B.2 donne un exemple simple d'interface de service, avec l'initialisation des stratégies d'information et de réaction via la méthode `launchReactions`. Cette méthode est appelée lorsque le *Environment Manager* détecte des situations dangereuses.

Programme B.2 – Interface du service proposé par le bundle *Reaction Manager*.

```
1 public interface IReactionsLauncherService {
2     public void launchReactions(FoundRiskConfigList configSet);
3 }
```

Le programme B.3 présente quant à lui la classe de définition de ce même service. On remarque que cette classe implémente effectivement l'interface d'accès au service, et vient définir le contenu de la méthode `launchReactions` en traitant séparément les différents contextes trouvés (méthode `splitAndTreat`).

Programme B.3 – Définition du service proposé par le bundle *Reaction Manager*.

```
1 public class ReactionsLauncherService implements IReactionsLauncherService{
2
3     private Receiver receiver;
4
5     public ReactionsLauncherService(Receiver receiver) {
6         this.receiver = receiver;
7     }
8
9     @Override
10    public void launchReactions(FoundRiskConfigList configSet) {
11        receiver.splitAndTreat(configSet);
12    }
13 }
```

Ce processus doit être réitéré pour chacun des services définis dans les composants OSGi de l'application.

B.2 Description des données d'entrée

La section 3.8 a indiqué la nature des informations d'entrée à fournir à l'application afin de pouvoir fonctionner. Ces informations doivent donc être transposées dans une syntaxe interprétable par INOVUS. Notre choix a été d'utiliser le langage XML afin de décrire ces données. Cette section propose donc deux exemples de représentation des données d'entrée, concernant tout d'abord l'environnement cible, puis la modélisation des contextes de risque que l'application doit couvrir.

B.2.1 Représentation de l'environnement

Le programme B.4 propose un extrait de la description de l'environnement de test utilisé dans ce prototype. Il s'agit de l'appartement du laboratoire Domus.

Le tableau B.1 détaille la grammaire utilisée pour décrire l'environnement cible.

Programme B.4 – Extrait de la représentation XML de l'appartement-test du laboratoire Domus.

```
1 <environment>
2   <stoveelements>
3     <stoveelement
4       elementType="hotplate" elementId="HTP1" elementName="Hotplate_1">
5     </stoveelement>
6     ...
7     <stoveelement elementType="oven"
8       elementId="OVEN" elementName="Oven">
9     </stoveelement>
10  </stoveelements>
11  <area areaId="KTN" areaName="Kitchen">
12    <sensors>
13      <sensor sensorId="KTN_POS"
14        sensorName="Kitchen_Position" stoveelement="">
15        <stateList>
16          <state parameterstate="OFF"></state>
17          <state parameterstate="ON"></state>
18        </stateList>
19      </sensor>
20      <sensor
21        sensorId="KTN_SMK" sensorName="Kitchen_Smoke" stoveelement="">
22        <stateList>
23          <state parameterstate="OFF"></state>
24          <state parameterstate="ON"></state>
```

B.2. DESCRIPTION DES DONNÉES D'ENTRÉE

```
25         </stateList>
26     </sensor>
27     <sensor sensorId="KTN_TMP_OVDOOR"
28         sensorName="Oven_Door_Temperature" stoveelement="OVEN">
29         <stateList>
30             <state parameterstate="COLD"></state>
31             <state parameterstate="HOT"></state>
32         </stateList>
33     </sensor>
34     <sensor sensorId="KTN_TMP_HTP1"
35         sensorName="Hotplate_l_Temperature" stoveelement="HTP1">
36         <stateList>
37             <state parameterstate="COLD"></state>
38             <state parameterstate="HOT"></state>
39         </stateList>
40     </sensor>
41     ...
42     <sensor sensorId="KTN_USE_HTP1"
43         sensorName="Hotplate_l_Use_Detector" stoveelement="HTP1">
44         <stateList>
45             <state parameterstate="OFF"></state>
46             <state parameterstate="ON"></state>
47         </stateList>
48     </sensor>
49     ...
50 </sensors>
51 <actuators>
52     <actuator actuatorId="KTN_SCR"
53         actuatorName="Kitchen_Screen" actuatorType="screen">
54     </actuator>
55     <actuator actuatorId="KTN_SPK"
56         actuatorName="Kitchen_Speaker" actuatorType="speaker">
57     </actuator>
58     <actuator actuatorId="KTN_LGHT_OVN"
59         actuatorName="Oven_Light" actuatorType="light">
60     </actuator>
61     <actuator actuatorId="KTN_LGHT_HTP1"
62         actuatorName="Hotplate_l_Light" actuatorType="light">
63     </actuator>
64     ...
65 </actuators>
66 </area>
67 ...
68 </environment>
```

La grammaire de description décrite ici (balises et attributs) doit être scrupuleusement respectée, quel que soit l'environnement étudié.

ANNEXE B. DÉTAILS SUR LE PROTOTYPE D'INOVUS VALIDANT DE L'APPROCHE DE DÉVELOPPEMENT

B.2.2 Description de situation à risque

De la même manière, le programme B.5 fournit un extrait de la description des contextes de risque, selon une grammaire préétablie et synthétisée par le tableau B.2. Là encore, cette grammaire doit être strictement suivie pour que l'application puisse charger les situations à couvrir.

Programme B.5 – Extrait de la représentation XML des contextes de risques.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <configurations>
4   <configuration configurationId="hot_surfaces" configurationType="Burns"
5     configurationName="Hot_surfaces_(Oven_door_or_hotplates)">
6     <parameters>
7       <parameter>
8         parameterId="KTN_TMP_HTP1" parameterCurrentValue="HOT"
9         parameterPreviousValue="" parameterTransitionTime="0">
10      </parameter>
11      ...
12    </parameters>
13  </configuration>
14  <configuration configurationId="no_surveillance" configurationType="Fires"
15    configurationName="No_surveillance_on_heating_food_(No_regular_check)">
16    ...
17  </configuration>
18  <configuration configurationId="nobody_at_home" configurationType="Fires"
19    configurationName="No_surveillance_on_heating_food_(No_regular_check)">
20    ...
21  </configuration>
22  <configuration configurationId="too_long_use" configurationType="Fires"
23    configurationName="Too_long_use_of_a_stove_element">
24    ...
25  </configuration>
26  <configuration configurationId="smoke_detection" configurationType="Intoxication"
27    configurationName="Smoke_detection_in_kitchen">
28    <parameters>
29      <parameter>
30        parameterId="KTN_SMK" parameterCurrentValue="ON"
31        parameterPreviousValue="" parameterTransitionTime="0">
32      </parameter>
33    </parameters>
34  </configuration>
35 </configurations>
```

Tableau B.1 – Grammaire de description de l'environnement cible.

Balise	Rôle
environment	Balise maître du document. Contient les stoveelements et les area.
stoveelements	Contient la liste des éléments de cuisson de l'environnement concerné.
stoveelement	Décrit un élément de cuisson à l'aide d'un elementType, d'un elementId, et d'un elementName.
area	Décrit une zone géographique de l'environnement à l'aide de sensors et d'actuators. Représentée par un areaId et un areaName.
sensors	Contient la liste des capteurs de l'area concernée.
sensor	Décrit un capteur à partir d'un sensorId, d'un sensorName, d'un stoveelement, et d'une liste d'états possibles stateList.
stateList	Contient la liste des états possibles pour le sensor concerné.
state	Indique un état possible du sensor concerné à l'aide d'un parameterstate.
actuators	Contient la liste des acteurs de l'area concernée.
actuator	Décrit un acteur à partir d'un actuatorId, d'un actuatorName, et d'un actuatorType.

Tableau B.2 – Grammaire de description des contextes de risque.

Balise	Rôle
configurations	Balise maître du document. Contient les différentes configuration que l'application doit couvrir.
configuration	Décrit un contexte de risque à l'aide d'un configurationId, d'un configurationType, et d'un configurationName. Englobe la liste des paramètres qui constituent la situation dangereuse.
parameters	Contient une liste des paramètres pour la configuration concernée.
parameter	Décrit un paramètre à l'aide d'un parameterId, d'un parameterCurrentValue, d'un parameterPreviousValue, et d'un parameterTransitionTime. Ces attributs permettent d'indiquer des valeurs pour deux états successifs du capteur, et ainsi permettre de détecter non pas un état "critique", mais une transition spécifique. On peut aussi spécifier une durée pour cette transition.

Bibliographie

- [1] PerCom 2011.
« 9th IEEE Pervasive Computing and Communication (PerCom) conference ».
Consultation le 07 Mars 2012.
<http://www.percom.org/2011/>.
- [2] B. ABDULRAZAK, B. CHIKHAOUI, C. GOUIN-VALLERAND et B. FRAIKIN.
« A standard ontology for smart spaces ».
International Journal of Web and Grid Services, 6(3):244–268, 2010.
- [3] J. R. ABRIAL.
Modeling in Event-B : System and Software Engineering.
Cambridge University Press, 2010.
- [4] U.S. Fire ADMINISTRATION.
« Development and Testing of Temperature Sensors for Preventing Cooking Fires on Glass Ceramic Electric Ranges ».
Consultation le 07 Mars 2012.
http://www.usfa.fema.gov/fireservice/research/other/other_stove.shtm.
- [5] M. AHRENS.
« Home Smoke Alarms : The Data as Context for Decision ».
Fire Technology, 44(1):313–327, 2008.
- [6] OSGi ALLIANCE.
« Open Services Gateway initiative ».
Consultation le 07 Mars 2012.
<http://www.osgi.org/>.

BIBLIOGRAPHIE

- [7] W. K. ANEMAET et M. E. MOFFA-TROTTER.
« Promoting Safety and Functions Through Home Assessments ».
Geriatric Rehabilitation, 15(1):26–55, 1999.
- [8] C. BAIER et J.-P. KATOEN.
Principles of Model Checking (Representation and Mind Series).
The MIT Press, 2008.
- [9] M. BAUDET, N. AMOURQUX et G. HOUIN.
« Intoxications accidentelles domestiques ».
EMC-Toxicologie-Pathologie, 1(2):29–34, 2004.
- [10] A. J. BHARUCHA, V. ANAND, J. FORLIZZI, M. A. DEW, C. F. REYNOLDS III,
S. STEVENS et H. WACTLAR.
« Intelligent Assistive Technology, Applications to Dementia Care : Current Capabilities, Limitations, and Future Challenges. ».
The American Journal of Geriatric Psychiatry, 17(2):88–104, 2009.
- [11] J. BISWAS, M. MOKHTARI, J. DONG et P. YAP.
« Mild dementia care at home—integrating activity monitoring, user interface plasticity and scenario verification ».
Aging Friendly Technology for Health and Independence, pages 160–170, 2010.
- [12] J. BOURGEOIS.
« *Vivre avec la démence à domicile : évaluation des situations à risque pour le patient et de la vigilance chez l'aidant informel.* ».
Thèse de doctorat, Université Pierre Mendès France - Grenoble II, 2009.
- [13] J. BOURGEOIS, P. COUTURIER et J. TYRRELL.
« Sécurité à domicile des personnes démentes : étude préliminaire sur les situations à risque en consultation mémoire de gériatrie en France. ».
Psychologie et neuropsychiatrie du vieillissement, 7(3):213–224, 2009.
- [14] P. BUSNEL, A. GHOURAF et H. PIGOT.
« Validation Process of an Activity of Daily Living Simulator ».
Dans *Proceedings of the 2006 Summer Computer Simulation Conference*, pages 356–362, 2006.
- [15] S. E. CARTER, E. M. CAMPBELL, R. W. SANSON-FISHER, S. REDMAN et W. J. GILLESPIE.

BIBLIOGRAPHIE

- « Environmental hazards in the homes of older people ».
Age and ageing, 26:195–202, 1997.
- [16] M. CHAN, D. ESTÈVE, C. ESCRIBA et E. CAMPO.
« A review of smart homes—Present state and future challenges ».
Computer Methods and Programs in Biomedicine, 91(1):55–81, 2008.
- [17] Z. CHAO.
« Fire Safety », 2004.
- [18] J. H. CHEN, P. P. Y. CHI, H. H. CHU, C. C. H. CHEN et P. HUANG.
« A Smart Kitchen for Nutrition-Aware Cooking ».
IEEE Pervasive Computing, 9(4):58–65, 2010.
- [19] Z. CHEN et S. FICKAS.
« Do No Harm : Model Checking eHome Applications ».
Dans *Proceedings of the 1st International Workshop on Software Engineering for Pervasive Computing Applications, Systems, and Environments*, page 8. IEEE Computer Society, 2007.
- [20] P. Y. CHI, J. H. CHEN, H. H. CHU et J. L. LO.
« Enabling calorie-aware cooking in a smart kitchen ».
Persuasive Technology, pages 116–127, 2008.
- [21] L. CLEMSON, M. H. FITZGERALD et R. HEARD.
« Content Validity of an Assessment Tool to Identify Home Fall Hazards : the Westmead Home Safety Assessment. ».
British Journal of Occupational Therapy, 62(4):171–179, 1999.
- [22] L. CLEMSON, L. MACKENZIE, C. BALLINGER, J.C.T. CLOSE et R.G. CUMMING.
« Environmental interventions to prevent falls in community-dwelling older people ».
Journal of Aging and Health, 20(8):954, 2008.
- [23] T. de CHAMPS, M. OUENZAR, B. ABDULRAZAK, M. FRAPPIER, H. PIGOT et B. FRAIKIN.
« Pervasive safety application with model checking in smart houses : The IN-OVUS intelligent oven ».
Dans *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2011 IEEE International Conference on, pages 630–635, march 2011.

BIBLIOGRAPHIE

- [24] G. DEMIRIS et B. K. HENSEL.
« Technologies for an aging society : A systematic review of “smart home” applications ».
Yearb Med Inform, pages 33–40, 2008.
- [25] Laboratoire DOMUS.
Consultation le 07 Mars 2012.
<http://domus.usherbrooke.ca/>.
- [26] J. J. DUNKIN et C. ANDERSON-HANLEY.
« Dementia caregiver burden ».
Neurology, 51(1 Suppl 1):S53, 1998.
- [27] N. DUTHEIL.
« Les aides et les aidants de personnes âgées. Etudes et résultats. ».
Direction de la Recherche des Études de l'Évaluation et des Statistiques, (142), 2001.
- [28] A. EGYED.
« Instant consistency checking for the UML ».
Dans *ICSE '06 : Proceedings of the 28th international conference on Software engineering*, pages 381–390. ACM, 2006.
- [29] A. R. EHRLICH, R. Y. BAK, P. WALD-CAGAN et D. F. GREENBERG.
« Risk factors for fires and burns in homebound, urban elderly ».
Journal of Burn Care & Research, 29(6):985, 2008.
- [30] A. R. EHRLICH, S. KATHPALIA, Y. BOYARSKY, A. SCHECHTER et P. BIJUR.
« Elderly patients discharged home from the emergency department with minor burns ».
Burns, 31(6):717–720, 2005.
- [31] HomeSense ENTERPRISES.
Consultation le 07 Mars 2012.
<http://www.homesensers.com/>.
- [32] StoveTop FIRESTOP.
Consultation le 07 Mars 2012.
<http://www.stovetopfirestop.com/>.

BIBLIOGRAPHIE

- [33] The Apache Software FOUNDATION.
« Apache Felix ».
Consultation le 07 Mars 2012.
<http://awarehome.imtc.gatech.edu/>.
- [34] P. FRADET, D. MÉTAYER et M. PÉRIN.
« Consistency Checking for Multiple View Software Architectures ».
Dans Oscar NIERSTRASZ et Michel LEMOINE, éditeurs, *Software Engineering — ESEC/FSE '99*, volume 1687 de *Lecture Notes in Computer Science*, pages 410–428. Springer Berlin / Heidelberg, 1999.
- [35] M. FRAPPIER, B. FRAIKIN, R. CHOSSART, R. CHANE-YACK-FA et M. OUENZAR.
« Comparison of Model Checking Tools for Information Systems ».
Dans *12th International Conference on Formal Engineering Methods (ICFEM 2010)*. Springer, 2010.
- [36] M. FRIAS, J. P. GALEOTTI, C. L. POMBO et N. AGUIRRE.
« DynAlloy : upgrading Alloy with actions ».
Dans Gruia-Catalin ROMAN, éditeur, *ICSE 2005 : 27th International Conference on Software Engineering*, pages 442–450. ACM Press, 2005.
- [37] Ronald L. GEORGE.
« Plumbing to Prevent Domestic Hot Water Scalds ».
Consultation le 07 Mars 2012.
<http://www.hgexperts.com/article.asp?id=5135>.
- [38] GERONTECHNOLOGIE.NET.
« Un indicateur de température d'eau au sortir du robinet ».
Consultation le 07 Mars 2012.
<http://www.gerontechnologie.net/indicateur-de-temperature-deau-au-sortir-du-robinet>.
- [39] L. D. GILLESPIE, W. J. GILLESPIE, M. C. ROBERTSON, S. E. LAMB, R. G. CUMMING et B. H. ROWE.
« Interventions for preventing falls in elderly people (Review). ».
The Cochrane Library, 4, 2003.

BIBLIOGRAPHIE

- [40] H. GILMOUR.
« Living alone with dementia : risk and the profesional role. ».
Nursing older people, 16(9):20–24, 2004.
- [41] K. Z. HAIGH et H. A. YANCO.
« Automation as Caregiver : a Survey of Issues and Technologies. ».
Dans *AAAI 02 Workshop "Automation as Caregiver"*, pages 39–53. American Association for Artificial Intelligence, 2002.
- [42] B. HAILPERN et P. SANTHANAM.
« Software debugging, testing, and verification ».
IBM Systems Journal, 41(1):4 – 12, 2002.
- [43] J. R. HALL.
« Home cooking fire patterns and trends ».
National Fire Protection, 2006.
- [44] R. D. HARPER et W. A. DICKSON.
« Reducing the burn risk to elderly persons living in residential care ».
Burns, 21(3):205–208, 1995.
- [45] C. L. HEITMEYER, R. D. JEFFORDS et B. G. LABAW.
« Automated consistency checking of requirements specifications ».
ACM Transactions on Software Engineering and Methodology, 5(3):231–261, 1996.
- [46] D. HERZBERG, N. MARSDEN, C. LEONHARDT, P. KUBLER, H. JUNG, S. THOMANEK et A. BECKER.
« Using formal specification techniques for advanced counseling systems in health care ».
HCI and Usability for Medicine and Health Care, pages 41–54, 2007.
- [47] A. J. HILL, F. GERMA et J. C. BOYLE.
« Burns in older people - outcomes and risk factors. ».
Journal of the American Geriatrics Society, 50(11):1912–1913, 2002.
- [48] C. HOAREAU et I. SATOH.
« A model checking-based approach for location query processing in pervasive computing environments ».

BIBLIOGRAPHIE

- Dans *On the Move to Meaningful Internet Systems 2007 : OTM 2007 Workshops*, pages 866–875. Springer, 2007.
- [49] Z. HUZAR, L. KUZNIARZ, G. REGGIO et J.-L. SOURROUILLE.
« Consistency Problems in UML-Based Software Development ».
Dans Nuno NUNES, Bran SELIC, Alberto Rodrigues da SILVA et Ambrosio TOVAL ALVAREZ, éditeurs, *UML Modeling Languages and Applications*, volume 3297 de *Lecture Notes in Computer Science*, pages 1–12. Springer Berlin / Heidelberg, 2005.
- [50] Protection E.M. INC..
Consultation le 07 Mars 2012.
<http://www.safecook.ca/>.
- [51] G. R. ISTRE et S. MALLONEE.
« Smoke alarms and prevention of house-fire—related deaths and injuries ».
Western Journal of Medicine, 173(2):92, 2000.
- [52] D. JACKSON.
Software Abstractions.
MIT Press, 2006.
- [53] Daniel JACKSON.
« Alloy Analyzer 4, ».
Consultation le 07 Mars 2012.
<http://alloy.mit.edu/alloy4/>.
- [54] K. KALLIN, J. JENSEN, L. L. OLSSÓN, L. NYBERG et Y. GUSTAFSON.
« Why the elderly fall in residential care facilities, and suggested remedies ».
The Journal of Family Practice, 53(1):41–527, 2004.
- [55] D. KILLALEA.
« Reducing Residential Fires Fatalities ».
Rapport Technique, Tasmania Fire Service, 1999.
- [56] C. LEFEBVRE et F. VÉZINA.
« Adapter l’environnement, un moyen pour aider à compenser les déficits cognitifs. ».
Le Médecin du Québec, 37(4):83–87, 2002.

- [57] L. LETTS, S. SCOTT, J. BURTNEY, L. MARSHALL et M. MCKEAN.
« The Reliability and Validity of the Safety Assessment of Function and the Environment for Rehabilitation (SAFER Tool). ».
British Journal of Occupational Therapy, 61(3):127–132, 1998.
- [58] R. C. LOU, A. GIULIANO et M. D. MULVENNA.
« State of the Art in Electronic Assistive Technologies for People with Dementia ».
Supporting People with Dementia Using Pervasive Health Technologies, pages 23–35, 2010.
- [59] Stove Guard International LTD..
Consultation le 07 Mars 2012.
<http://stoveguard.ca/>.
- [60] T. J. LUI.
« Automation in Home Appliances ».
Springer Handbook of Automation, pages 1469–1483, 2009.
- [61] SCD LUISINA.
« Luisina Design ».
Consultation le 07 Mars 2012.
<http://www.luisina.com/>.
- [62] L. MACKENZIE, J. BYLES et N. HIGGINBOTHAM.
« Designing the Home Falls and Accidents Screening Tool (HOME FAST) : Selecting the Items. ».
British Journal of Occupational Therapy, 63(6):260–269, 2000.
- [63] MAIF.
« Le Maintien à Domicile ».
Consultation le 07 Mars 2012.
<http://www.le-maintien-a-domicile.fr/>.
- [64] S. MALIK et L. ZHANG.
« Boolean satisfiability from theoretical hardness to practical success ».
Communications of the ACM, 52(8):76–82, 2009.

BIBLIOGRAPHIE

- [65] S. MALLONEE, G. R. ISTRE, M. ROSENBERG, M. REDDISH-DOUGLAS, F. JORDAN, P. SILVERSTEIN et W. TUNELL.
« Surveillance and Prevention of Residential-Fire Injuries ».
The New England Journal of Medicine, 335(1):27–31, 1996.
- [66] F. J. M. MEILAND, J. A. C. DANSE, J. F. WENDTE, N. S. KLAZINGA et L. J. GUNNING-SCHEPERS.
« Caring for relatives with dementia—caregiver experiences of relatives of patients on the waiting list for admission to a psychogeriatric nursing home in The Netherlands ».
Scandinavian journal of public health, 29(2):113, 2001.
- [67] B. MEYER.
« On Formalism in Specifications ».
IEEE Software, 2(1):6–26, 1985.
- [68] F. G. MISKELLY.
« Assistive technology in elderly care. ».
Age and ageing, 30:455–458, 2001.
- [69] MOBILE et Pervasive Computing RESEARCH.
University of Florida, Consultation le 07 Mars 2012.
<http://www.icta.ufl.edu/gt.htm9>.
- [70] R. O. MORGAN, C. A. DEVITO, J. A. STEVENS, C. M. BRANCHE, B. A. VIRNIG, P. A. WINGO et R. W. SATTIN.
« A self-assessment tool was reliable in identifying hazards in the homes of elders ».
Journal of clinical epidemiology, 58(12):1252–e1, 2005.
- [71] B. NUSEIBEH, S. EASTERBROOK et A. RUSSO.
« Making inconsistency respectable in software development ».
Journal of Systems and Software, 58(2):171–180, 2001.
- [72] Georgia Institute of TECHNOLOGY.
« Aware Home Research Initiative ».
Consultation le 07 Mars 2012.
<http://awarehome.imtc.gatech.edu/>.

- [73] Joint Commission on Accreditation of HEALTHCARE ORGANIZATIONS.
Consultation le 07 Mars 2012.
<http://www.jointcommission.org/>.
- [74] M. PEZZÈ et M. YOUNG.
Software Testing and Analysis : Process, Principles, and Techniques.
Wiley, 2008.
- [75] S. L. PFLEEGER et J. M. ATLEE.
Software Engineering : Theory and Practice.
Pearson Prentice Hall, 3 édition, 2006.
- [76] High Tech PLACE.
« Embout robinet lumineux ».
Consultation le 07 Mars 2012.
<http://www.hightechplace.com/electronique/eclairage-led/ampoules-led.html>.
- [77] L. Poulin de COURVAL, I. GÉLINAS, S. GAUTHIER, D. GAYTON, L. LIU, M. ROSSIGNOL, J. SAMPALIS et D. DASTOOR.
« Reliability and Validity of the Safety Assessment Scale for people with dementia living at home. ».
The Canadian Journal of Occupational Therapy, 73(2):67–75, 2006.
- [78] L. D. PROCKOP et R. I. CHICHKOVA.
« Carbon monoxide intoxication : an updated review ».
Journal of the neurological sciences, 262(1-2):122–130, 2007.
- [79] F. PUISIEUX, V. PARDESSUS et S. BOMBOIS.
« Démences et chutes, deux problèmes liés chez la personne âgée ».
Psychologie et neuropsychiatrie du vieillissement, 3(4):271–279, 2005.
- [80] R. H. ROBNETT, V. HOPKINS et J. G. KIMBALL.
« The SAFE AT HOME : A quick home safety assessment ».
Physical & Occupational Therapy in Geriatrics, 20(3-4):77–101, 2003.
- [81] Y. ROLLAND, S. ANDRIEU, C. CANTET, J. E. MORLEY, D. THOMAS, F. NOURHASHEMI et B. VELLAS.
« Wandering behavior and Alzheimer disease. The REAL. FR prospective study ».
Alzheimer Disease & Associated Disorders, 21(1):31, 2007.

BIBLIOGRAPHIE

- [82] Link Care SERVICES.
« EDAO ».
Consultation le 07 Mars 2012.
<http://www.edao.com/domicile/edao-system/>.
- [83] Alzheimer's SOCIETY.
Consultation le 07 Mars 2012.
<http://www.alzheimers.org.uk/>.
- [84] A. SORCINELLI, L. SHAW, A. FREEMAN et K. COOPER.
« Evaluating The Safe Living Guide : A Home Hazard Checklist for Seniors ».
Canadian Journal on Aging, 26(2):127–137, 2007.
- [85] E. K. TANNER.
« Assessing Home Safety in Homebound Older Adults ».
Geriatric Nursing, 24(4):250–256, 2003.
- [86] K. M. THOM et S. E. E. BLAIR.
« Risk in Dementia - Assessment and Management : a Literature Review. ».
British Journal of Occupational Therapy, 61(10):441–447, 1998.
- [87] P. THOMAS, C. HAZIF-THOMAS, V. DELAGNES, P. BONDUELLE et J.-P. CLÉMENT.
« La vulnérabilité de l'aidant principal des malades déments à domicile. L'étude Pixel. ».
Psychologie et neuropsychiatrie du vieillissement, 3(3):207–220, 2005.
- [88] J. TILLY et P. REED.
« Falls, wandering, and physical restraints : a review of interventions for individuals with dementia in assisted living and nursing homes ».
Alzheimer's Care Today, 9(1):45, 2008.
- [89] TMIO.
« Professional cooking solutions for your kitchen ».
Consultation le 07 Mars 2012.
<http://www.tmio.com/products/>.
- [90] A. WAI, S. SHANTHINI DEVI, J. BISWAS et S. PANDA.
« Pervasive Intelligence System to Enable Safety and Assistance in Kitchen for

BIBLIOGRAPHIE

Home-Alone Elderly ».

Toward Useful Services for Elderly and People with Disabilities, pages 276–280, 2011.

[91] L. J. WARDA et M. F. BALLESTEROS.

« Interventions to prevent residential fire injury ».

Handbook of injury and violence prevention, pages 97–115, 2007.

[92] R. WATSON et S. M. GREEN.

« Feeding and dementia : a systematic literature review. ».

Journal of Advanced Nursing, 54(1):86–93, 2006.