



HAL
open science

Contribution à la Modélisation et à la Gestion des Interactions Produit-Processus dans la Chaîne Logistique par l'Approche Produits Communicants

Aldo Cea Ramirez

► **To cite this version:**

Aldo Cea Ramirez. Contribution à la Modélisation et à la Gestion des Interactions Produit-Processus dans la Chaîne Logistique par l'Approche Produits Communicants. Automatique / Robotique. Université Henri Poincaré - Nancy I, 2006. Français. NNT: . tel-00139831

HAL Id: tel-00139831

<https://theses.hal.science/tel-00139831v1>

Submitted on 3 Apr 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



FACULTE DES SCIENCES & TECHNIQUES

U.F.R. Sciences et Techniques Mathématiques, Informatique et Automatique
Ecole Doctorale IAEM Lorraine
Département de Formation Doctorale Automatique et Production Automatisée

Thèse

Présentée pour l'obtention du titre de

Docteur de l'Université Henri Poincaré, Nancy-I

en Automatique, Traitement du Signal, Génie Informatique

par **Aldo Alexis CEA RAMIREZ**

**Contribution à la Modélisation et à la Gestion des Interactions
Produit-Processus dans la Chaîne Logistique
par l'Approche Produits Communicants**

Soutenue publiquement le 18 juillet 2006

Membres du jury :

Rapporteurs :	M. Jean-Pierre CAMPAGNE	Professeur, INSA Lyon - Président du jury
	M. Duncan McFARLANE	Professeur, Université de Cambridge
	M. Jean Jacques PANSIOT	Professeur, Université Louis Pasteur Strasbourg
Examineurs :	M. Eddy BAJIC	Professeur, UHP, Nancy I (Directeur de recherche)
	M. Gérard MOREL	Professeur, UHP, Nancy I
	M. Didier WOLF	Professeur, INPL Nancy

Remerciements

Les travaux de recherche présentés dans ce mémoire ont été développés au Centre de Recherche en Automatique de Nancy (CRAN, UMR 7039 du CNRS) dans le cadre du projet SCP (Systèmes Contrôlés par le Produit) du groupe de recherche SYMPA (Systèmes de Production Ambiants), sous la direction du Professeur Eddy BAJIC.

Je tiens à exprimer mes plus sincères remerciements aux personnes qui ont rapporté ces travaux : Monsieur Jean-Pierre CAMPAGNE, Professeur à l'INSA Lyon, Monsieur Duncan McFARLANE, Directeur de recherche à l'Université de Cambridge (Angleterre) et Monsieur Jean Jacques PANSIOT, Professeur à l'Université Louis Pasteur Strasbourg.

Je remercie très vivement Monsieur Didier WOLF, Professeur à l'INPL Nancy, ainsi que Monsieur Gérard MOREL, Professeur à l'Université Henri Poincaré Nancy 1, pour l'intérêt qu'ils ont porté à ces travaux en tant qu'examineurs.

Je veux aussi exprimer toute ma reconnaissance à Monsieur Eddy BAJIC pour la confiance qu'il m'a accordée pendant le déroulement de mes travaux. Il m'a toujours écouté et encouragé dans mes travaux de recherche. Je remercie également l'ensemble des membres du CRAN pour leurs conseils et leur soutien.

Je remercie particulièrement mon université employeur au Chili (UVM - Universidad de Viña del Mar), la Commission Nationale de Recherche Scientifique et Technologique du Chili (CONICYT), le Gouvernement du Chili, et l'Ambassade de France au Chili pour le soutien financier offert à mes travaux de recherche.

Spécialement, je dirige mes plus chaleureux remerciements à Madame Elisabeth BECKER pour avoir accepté l'invitation à corriger la rédaction et l'orthographe de ce manuscrit.

Je tiens à remercier infiniment mon épouse et mes enfants qui m'ont accompagné dans cette aventure, en France, en me donnant tout leur amour. Bien entendu, un très grand merci à mes parents pour leur aide sentimentale et matérielle.

Table des matières

1	INTRODUCTION	1
1.1	La chaîne logistique et la gestion du cycle de vie d'un produit.....	1
1.2	Nouvelles tendances dans le domaine de la chaîne logistique.....	2
1.2.1	Produits et objets communicants	3
1.2.2	Intelligence ambiante.....	4
1.3	Les objets communicants dans des projets de recherche	5
1.4	Problématique scientifique.....	6
1.5	Structure de la thèse.....	7
2	ETAT DE L'ART SUR LES OBJETS COMMUNICANTS	9
2.1	Introduction aux objets communicants.....	9
2.1.1	Le paradigme <i>Ubiquitous Computing</i>	11
2.1.2	Le paradigme <i>Pervasive Computing</i>	13
2.1.3	Le concept de <i>Smart Object</i>	14
2.1.4	Le concept de Produit Intelligent.....	17
2.1.4.1	L'influence du paradigme HMS (<i>Holonic Manufacturing System</i>)	19
2.1.5	Synthèse des concepts	20
2.2	Classification des objets communicants.....	21
2.2.1	L'objet porteur de données.....	21
2.2.2	L'objet pointeur vers un système d'information.....	23
2.2.2.1	L'initiative <i>EPC Global</i>	24
2.2.2.2	L'objet Internet	26
2.2.2.3	L'objet Augmenté.....	27
2.2.3	L'objet comme prestataire et/ou demandeur de services	28
2.2.4	L'objet communicant sensitif	31
2.3	La technologie d'identification automatique pour les objets communicants.....	32
2.3.1	La technologie RFID	32
2.3.2	L'étiquette électronique ou tag	33
2.3.2.1	Étiquettes passives et étiquettes actives	35
2.3.2.2	Schéma fonctionnel d'une étiquette électronique.....	35
2.3.2.3	Modes de fonctionnement d'une étiquette : mode de lecture / écriture	36
2.3.3	Distance de communication	38
2.3.4	Fréquences de communication.....	38
2.3.5	Temps de communication	39
2.3.6	Méthodes de communication	40
2.3.7	Modèle de communication en couche pour la RFID.....	42
2.3.8	Standards pour la RFID.....	43
2.3.9	Contraintes dans l'usage des étiquettes radio fréquence	45
2.4	La technologie RFID et les objets communicants	46
2.5	Conclusions	47
3	ARCHITECTURES DE SERVICES AMBIANTS	49
3.1	Introduction aux architectures de services ambiants	49
3.1.1	La notion d'informatique ambiante	49
3.1.2	Le concept de réseau ambiant	50
3.1.3	Services Ambiants	52
3.1.4	Paradigmes de communication	53
3.1.4.1	Paradigme de communication appliqué au domaine de la chaîne logistique.....	54
3.2	Caractéristiques d'une architecture de services ambiants.....	56

3.2.1	L'identification automatique d'un dispositif.....	56
3.2.2	La localisation d'un dispositif.....	56
3.2.3	La découverte automatique des services	57
3.2.3.1	Mode centralisé.....	57
3.2.3.2	Mode distribué.....	58
3.2.3.3	Mode centralisé versus mode distribué	58
3.2.4	La description d'un dispositif et de ses services.....	59
3.2.5	L'invocation des services	60
3.2.6	La notification des événements.....	60
3.3	Architectures de gestion de services ambiants.....	61
3.3.1	Jini (<i>Java intelligent network infrastructure</i>)	62
3.3.2	UPnP (<i>Universal Plug and Play</i>).....	63
3.3.3	OSGi (<i>Open Services Gateway Initiative</i>)	65
3.3.4	CORBA (<i>Common Object Request Broker Architecture</i>).....	66
3.3.5	Les Services Web	67
3.4	Analyse et synthèse comparative des architectures de services ambiants présentées	69
3.5	Analyse de l'architecture <i>Universal Plug and Play</i> (UPnP).....	74
3.5.1	Entité Dispositif UPnP	74
3.5.1.1	Services, actions, variables d'état et événements	74
3.5.2	Entité Point de contrôle UPnP	75
3.5.3	Entité mixte : Dispositif UPnP et Point de Contrôle UPnP	75
3.5.4	Modèle de la pile des protocoles réseau dans UPnP	75
3.5.4.1	HTTP <i>multicast</i> et HTTP <i>unicast</i>	76
3.5.4.2	XML (<i>Extensible Markup Language</i>).....	77
3.5.4.3	SSDP (<i>Simple Service Discovery Protocol</i>)	77
3.5.4.4	SOAP (<i>Simple Object Access Protocol</i>).....	78
3.5.4.5	GENA (<i>Generic Event Notification Architecture</i>)	78
3.5.5	Les fonctionnalités et mécanismes d'UPnP	78
3.5.5.1	L'adressage IP.....	80
3.5.5.2	L'annonce et la découverte d'un dispositif et ses services	80
3.5.5.3	La description d'un dispositif et ses services.....	80
3.5.5.4	Le contrôle d'un dispositif.....	82
3.5.5.5	La gestion des événements.....	83
3.5.5.6	La présentation d'un dispositif.....	83
3.5.6	Interactions entre un dispositif et un point de contrôle	83
3.6	Modèle conceptuel d'un objet communicant	85
3.7	Conclusions	87
4	PROPOSITION METHODOLOGIQUE	89
4.1	Présentation de l'approche proposée	89
4.1.1	Le produit physique et ses services dans son cycle de vie.....	89
4.2	Proposition méthodologique : types d'objets communicants	91
4.2.1	Cas 1 : Objet communicant passif	91
4.2.2	Cas 2 : Objet communicant actif.....	95
4.2.3	Cas 3 : Objet communicant actif unifié.....	96
4.3	Modélisation des services pour un objet communicant.....	98
4.3.1	Facteurs déterminant les services des objets communicants	98
4.3.1.1	Nature de l'objet physique	98
4.3.1.2	Domaines d'interaction des objets communicants	100
4.3.1.3	Profil d'acteur de la chaîne logistique.....	101
4.3.2	Définition des services d'un objet communicant.....	102

4.3.2.1	Définition des variables d'état	104
4.3.2.2	Définition des événements	104
4.3.2.3	Définition des actions	104
4.3.3	Documents XML décrivant l'objet communicant	105
4.3.4	Classes de services pour un objet communicant	106
4.4	Modélisation des interactions entre objets communicants et acteurs de la chaîne logistique	110
4.4.1	Annonce d'un objet communicant et de ses services dans un réseau ambiant	111
4.4.2	Découverte d'un objet communicant et de ses services	111
4.4.3	Description des services d'un objet communicant	112
4.4.4	Invocation des services d'un objet communicant par un dispositif	113
4.4.5	Invocation des services d'un dispositif par un objet communicant actif	114
4.4.6	Invocation de services d'un objet communicant par un objet communicant actif	115
4.4.7	Souscription aux événements d'un objet communicant dans un réseau ambiant	116
4.5	Bilan de la méthodologie proposée	117
4.6	Conclusions	119
5	APPLICATION DE LA METHODOLOGIE : CAS D'ETUDE D'UN ENTREPOT	121
5.1	Description du cas d'étude	121
5.2	Architecture de Services proposée pour un entrepôt	122
5.2.1	Facteurs déterminant les services d'un produit ou objet communicant	123
5.2.1.1	Services dépendant de la famille et type d'objet physique	123
5.2.1.2	Services dépendants du domaine d'interaction	124
5.2.1.3	Services dépendant du profil de l'acteur de l'entrepôt	124
5.3	Classification des services offerts par les acteurs de l'entrepôt	124
5.3.1	Classe de service « Identification d'un acteur »	125
5.3.2	Classe de service « Description d'un acteur »	125
5.3.3	Classe de service « Etat d'un acteur »	125
5.3.4	Classe de service « Traçabilité d'un acteur »	126
5.3.5	Classe de service « Information augmentée de l'acteur »	126
5.4	Relations entre les acteurs de l'entrepôt par les services	128
5.4.1	Relation entre un objet communicant actif et le gestionnaire de l'entrepôt UPnP	129
5.4.2	Relation entre un objet communicant actif et les moyens de transports UPnP	131
5.4.3	Relation entre un objet communicant actif et capteur UPnP	133
5.5	Définition des services offerts par les produits communicants	134
5.5.1	Type de service « Identification d'un produit communicant »	134
5.5.2	Type de service « Conditions de stockage d'un produit communicant »	135
5.5.3	Type de Service « Etats d'un produit communicant »	137
5.5.4	Type de service « Traçabilité d'un produit communicant »	138
5.5.5	Type de service « Information augmentée d'un produit communicant »	139
5.6	Interactions dans le système par les services	139
5.6.1	Entrée et stockage des produits communicants	139
5.6.2	Transport des produits communicants dans l'entrepôt	140
5.6.3	Gestion des alarmes dans l'entrepôt	141
5.6.4	Sortie des produits communicants de l'entrepôt	142
5.7	Génération de démonstrateur de laboratoire	142
5.7.1	Démonstrateur objet communicant passif / actif	143
5.7.1.1	Création et description des services de l'objet communicant	144
5.7.1.2	Génération de l'application logiciel	145
5.7.2	Démonstrateur point de contrôle	147
5.7.3	Démonstrateur capteur UPnP	148
5.8	Conclusion	148

6	CONCLUSIONS DE LA THESE	151
7	REFERENCES.....	153

Table des illustrations

Figure 1 : Exemple générique d'une chaîne Logistique	1
Figure 2 : Les capacités liées à un Objet Communicant.....	9
Figure 3 : Exemples de dispositifs Objets Communicants.....	10
Figure 4 : Exemples d'interactions ubiquistes entre des utilisateurs et des objets.....	12
Figure 5: Interactions d'éléments clés selon le paradigme <i>Ubiquitous Computing</i>	13
Figure 6 : Exemples de Smarts Objects : l'objet <i>Mediacup</i> et le composant intégré Smart-its.	15
Figure 7 : Le concept de « smart box » et l'application <i>Smart Medicine Cabinet</i>	16
Figure 8 : Apport de l'information du produit dans le processus décisionnelle.....	18
Figure 9 : Optimisation d'un système de contrôle par les informations stockées sur le produit et l'information captée de l'environnement.	19
Figure 10 : Approche produit / objet intelligent portant sa propre information.....	20
Figure 11 : Exemple de codes à barre à deux dimensions.....	22
Figure 12 : Des formats classiques de codes à barre linéaires employés dans la chaîne logistique.	22
Figure 13 : L'objet pointe vers ses informations de recyclage accessibles par une page Web.....	23
Figure 14 : Approche objet comme vecteur d'information	24
Figure 15 : Code EPC de 96 bits.....	25
Figure 16 : Architecture EPC d'association de système d'information au Produit.....	25
Figure 17 : Le système Dialog.....	26
Figure 18 : Utilisateur obtenant de l'information augmentée d'un poster intelligent.....	28
Figure 19 : L'objet comme demandeur ou fournisseur de services.....	30
Figure 20 : Un « <i>sentient artefact</i> » : le miroir sensitif.....	32
Figure 21 : Structure générale d'un système RFID.....	33
Figure 22 : Formes d'étiquettes électroniques, exemple produits Omron.....	34
Figure 23 : Vue du silicium d'une étiquette électronique (photo Philips Semiconductors).....	34
Figure 24 : Communication Lecteur – Etiquette par couplage magnétique.....	35
Figure 25 : Schème fonctionnel d'un transpondeur passif avec mémoire intégré.....	36
Figure 26 : Transpondeur avec un capteur de température intégré.....	36
Figure 27 : Temps de transaction Lecteur – Etiquette selon les données accédées.....	40
Figure 28 : Méthodes d'accès aux étiquettes électroniques.....	41
Figure 29 : Lobe de rayonnement d'une antenne RFID 13.56 Mhz.....	41
Figure 30 : Temps de communication avec des étiquettes électronique radiofréquences.....	42
Figure 31 : Modèle de communication RFID basé sur le modèle OSI.....	43
Figure 32 : Normes RFID et bandes de fréquences.....	45
Figure 33 : Influence du positionnement Etiquette / Antenne.....	45
Figure 34 : Influence de l'environnement métallique sur l'accès à une étiquette.....	46
Figure 35 : Domaines et ressources du concept d'informatique ambiante.....	50
Figure 36 : Sphères d'influence dans un réseau ambiant.....	51
Figure 37 : Liaison entre différents types de réseaux ambiants.....	51
Figure 38 : Domaines de services dans un réseau ambiant.....	53
Figure 39 : Représentation du paradigme client/serveur et du paradigme souscripteur/producteur.....	54
Figure 40 : Mode de communication « <i>request / reply</i> » : le produit / objet communicant comme fournisseur et demandeur de services	55
Figure 41 : Mode de communication « <i>souscription / notification</i> » : le produit intelligent comme annonceur et souscripteur d'événements.....	55
Figure 42 : Enregistrement et découverte active des services en mode centralisé.....	57
Figure 43 : Découverte des services en mode distribué.....	58

Figure 44 : Description standardisée d'un dispositif et de ses services.....	59
Figure 45 : Diagramme d'échange pour l'invocation de service en mode distribué.....	60
Figure 46 : Diagramme d'échange de souscription et notification vers un dispositif.....	61
Figure 47 : Fonctionnement générale de Jini.....	62
Figure 48 : Points de contrôle UPnP en invoquant des services vers des dispositifs UPnP.....	63
Figure 49 : Architecture OSGi.....	65
Figure 50 : Architecture CORBA et Modèle <i>Object Management Architecture</i> (OMA).....	67
Figure 51 : Fonctionnement générale des Services Web.....	68
Figure 52 : Exemple d'une architecture réseaux P2P distribuée.....	70
Figure 53 : Diagramme de classe décrivant un dispositif UPnP.....	75
Figure 54 : Pile de protocoles réseau dans UPnP.....	76
Figure 55: Modèle d'un Dispositif <i>Device</i> UPnP.....	79
Figure 56 : Modèle d'un Point de contrôle UPnP.....	79
Figure 57 : L'annonce d'un dispositif et la recherche des dispositifs dans un réseau ambiant UPnP... ..	80
Figure 58 : Fichier XML de description d'un dispositif.....	81
Figure 59 : Fichier XML de description des services d'un dispositif.....	82
Figure 60 : L'invocation d'une action sur le dispositif.....	82
Figure 61 : La souscription aux événements.....	83
Figure 62 : Diagramme de séquence décrivant la recherche d'un dispositif et l'invocation d'une action.	84
Figure 63 : Diagramme de séquence d'annonce d'un dispositif sur le réseau et souscription aux événements.....	84
Figure 64 : Modèle conceptuel d'un produit communicant.....	85
Figure 65 : Interactions entre un produit communicant, des processus et des utilisateurs au travers des services dans la chaîne logistique.....	90
Figure 66 : Objet communicant passif dans un réseau ambiant UPnP.....	92
Figure 67 : Principe de triangulation appliqué à la localisation des objets communicants.....	93
Figure 68 : Objet communicant actif dans un réseau ambiant UPnP.....	95
Figure 69 : Objet communicant actif unifié.....	97
Figure 70 : Exemple de décomposition de familles d'un objet.....	99
Figure 71: Domaines d'interactions pour un objet communicant.....	100
Figure 72 : Moyen de transport des objets communicants et ses rôles.....	101
Figure 73 : Dépendances des services d'un objet communicant.....	102
Figure 74 : Diagramme de classes UML décrivant statiquement un objet communicant.....	103
Figure 75 : Document XML décrivant globalement un objet communicant.....	106
Figure 76 : Document XML décrivant un service d'un objet communicant.....	106
Figure 77 : Déroulement du Cycle de vie et Classes de Services associées des Produits Communicants.	107
Figure 78 : Domaines géographiques dans un domaine fonctionnel d'un objet communicant.....	109
Figure 79 : Diagramme de séquence UML : Annonce d'un objet communicant passif sur le réseau UPnP.....	111
Figure 80 : Diagramme de séquence UML : Découverte d'un objet communicant passif par un point de contrôle UPnP.....	112
Figure 81 : Diagramme de séquence UML : Découverte d'un objet communicant passif par un objet communicant actif.....	112
Figure 82 : Diagramme de séquence UML : Description des services d'un objet communicant passif.	113
Figure 83 : Diagramme de séquence UML : Invocation d'action sur un objet communicant passif... ..	114
Figure 84 : Diagramme de séquence UML : Invocation d'action réalisée par un objet communicant actif.....	115

Figure 85 : Diagramme de séquence UML : Invocation d'actions entre objets communicants.....	116
Figure 86 : Diagramme de séquence UML : Souscription aux événements d'un produit communicant.	117
Figure 87 : Exemple de disposition des ressources dans un entrepôt.	121
Figure 88 : Architecture de Services proposée pour un entrepôt.	122
Figure 89 : Diagramme de relations entre les acteurs de l'entrepôt.	128
Figure 90 : Diagramme de cas d'utilisation UML décrivant l'usage du gestionnaire de l'entrepôt par l'objet communicant actif.	130
Figure 91 : Diagramme de séquence UML décrivant le mécanisme de gestion de priorités	131
Figure 92 : Diagramme de séquence UML décrivant le mécanisme pour établir un contrat de travail entre un demandeur et un fournisseur de services.	132
Figure 93 : Diagramme de cas d'utilisation UML	132
Figure 94 : Diagramme de cas d'utilisation UML	133
Figure 95 : Diagramme de séquence UML décrivant les interactions entre acteurs afin de stocker un objet physique dans l'entrepôt.....	140
Figure 96 : Diagramme de séquence décrivant les interactions entre acteurs afin de transporter un objet physique dans l'entrepôt.	141
Figure 97 : Diagramme de séquence UML décrivant les interactions entre un objet communicant actif et un capteur dans l'entrepôt.	141
Figure 98 : Diagramme de séquence UML décrivant les interactions entre les acteurs afin de gérer la sortie des objets physiques de l'entrepôt.	142
Figure 99 : Architecture de démonstration des interactions entre acteurs par les services.	143
Figure 100 : Mémoire d'une étiquette électronique de produit communicant.	144
Figure 101 : Diagramme XML décrivant le service « Composition d'une boîte ».....	145
Figure 102 : Dispositif ou <i>Device Point</i> d'un Objet Communicant Passif / Actif.	146
Figure 103 : Point de Contrôle UPnP qui permet d'interagir avec l'objet communicant.....	147
Figure 104 : Dispositif ou <i>Device Point</i> d'un capteur de température et humidité UPnP.	148

Table des tableaux

Tableau 1 : Des questions pour identifier l'information contextuelle d'un objet communicant.....	11
Tableau 2 : Résumé des principales performances des différentes fréquences RFID.	39
Tableau 3 : Temps de communication de tag MIFARE de Philips.....	40
Tableau 4 : Critères d'évaluation des architectures de services ambiants.	72
Tableau 5 : Synthèse comparative des architectures de services étudiées.....	73
Tableau 6 : Description des services d'un objet communicant.	104
Tableau 7 : Exemple de description de variables d'état pour un service « ALPHA ».	104
Tableau 8 : Exemple de description de actions à exécuter pour un service déterminé « ALPHA ».	105
Tableau 9 : Synthèse et implémentations des capacités d'un Produit Communicant	118
Tableau 10 : Relations entre acteurs par les services dans l'entrepôt.	129
Tableau 11 : Liste des services pour identifier un produit communicant	134
Tableau 12 : Variables décrivant les services pour identifier un produit communicant.....	134
Tableau 13 : Actions permettant l'identification d'un produit communicant.	135
Tableau 14 : Liste de service décrivant les conditions de stockage d'un objet physique.	135
Tableau 15 : Variables décrivant les conditions de stockage d'un objet physique.	136
Tableau 16 : Actions permettant d'obtenir les conditions de stockage d'un objet physique.	136
Tableau 17 : Liste de services servant à informer et modifier l'état d'un produit communicant.....	137
Tableau 18 : Variables décrivant les services pour identifier les états des produits communicants....	137
Tableau 19 : Actions sur les états des produits communicants.	137
Tableau 20 : Les services de domaine fonctionnel et géographique d'un produit communicant.	138
Tableau 21 : Variables des domaines fonctionnel et géographique du produit communicant	138
Tableau 22 : Actions des services de traçabilité de l'objet communicant.	139

1 Introduction

1.1 La chaîne logistique et la gestion du cycle de vie d'un produit

D'après le dictionnaire [APICS, 1998]¹ la chaîne logistique (*Supply Chain*) est « *the processes from the initial raw materials to the ultimate consumption of the finished product linking across supplier-user companies* ». Le concept de chaîne logistique a été actualisé et redéfini par l'APCIS en 2001 comme : « *the global network used to deliver products and services from raw materials to end customers through engineered flows of information, physical distribution, and cash* » [APICS, 2001]. Selon [Panozzo et al., 1999] la logistique est l'art de coordonner et de gérer les mouvements de produits et d'informations du point d'origine au point de destination final. Dans nos travaux, la chaîne logistique est vue comme un ensemble d'entreprises directement impliquées dans le cycle de vie d'un produit. Nous sommes intéressés par les flux physique et informationnel qui relient ces entreprises. La Figure 1 illustre le flux bidirectionnel d'information entre les acteurs de la chaîne logistique (Fournisseur A, Producteur A, Centre de distribution A et Détaillant A) et le flux physique unidirectionnel depuis le fournisseur vers le détaillant. On observe que la phase de transport doit gérer simultanément le flux d'information et le flux physique. La gestion de l'information et du flux physique doit être appuyée par des technologies de l'information et de la communication (TICs). Dans la figure, on observe une communication directe entre les participants. Il est possible aussi de réaliser la communication entre les parties au moyen d'un gestionnaire central de l'information.

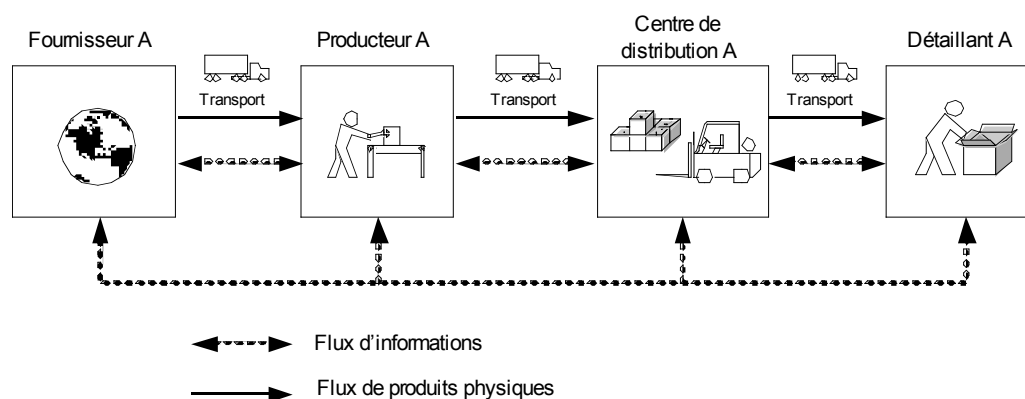


Figure 1 : Exemple générique d'une chaîne Logistique

Selon [Croom et al., 2000] la gestion de la chaîne logistique est un domaine multidisciplinaire regroupant des disciplines telles que la gestion d'opérations, les technologies de l'information et la communication, la gestion organisationnelle, la gestion de relations avec les clients, etc. Ceci a pour but global l'optimisation de la qualité de service offerte aux clients intermédiaires et finaux (représenté par l'opportunité des échanges et le rapport qualité / prix), tout en assurant la meilleure rentabilité économique pour chaque entreprise participante dans cette chaîne de activités. Dans ce cas, nous pouvons dire que la chaîne logistique représente une stratégie d'entreprises regroupant différentes vues d'une organisation. Selon [Tan, 2001] la gestion de la chaîne logistique intègre deux perspectives complémentaires² : D'une part, la perspective d'achat et d'approvisionnement et, d'autre part, la perspective de transport et logistique. La perspective d'achat

¹ <http://www.apics.org> - American Production and Inventory Control Society

² Pour l'auteur la notion de perspective de la chaîne logistique constitue une vue particulière du processus global de gestion des produits dans son cycle de vie.

et d'approvisionnement de ressources (matériels, pièces, ...) entre entreprises partenaires permet, à court terme, d'augmenter la productivité, de réduire le niveau de stock, de réduire les temps de cycle, et à long terme, d'augmenter la satisfaction des clients et d'accroître les bénéfices des membres de cette communauté virtuelle. Cette perspective est très centrée sur la transformation du produit, principalement dans les industries manufacturières. La perspective « transport et logistique » définit un système intégrant fournisseurs, producteurs, moyens de transport, entrepôts, centres de distribution, et points de vente permettant d'apporter une information correcte, où et quand nécessaire, afin de réduire l'incertitude, augmenter la visibilité des produits et augmenter l'efficacité globale de la chaîne de participants. Ainsi, la fusion de ces deux perspectives permet d'intégrer les fonctions de l'industrie manufacturière aux fonctions classiques de la logistique dans une chaîne logistique intégrale.

Selon [Lejeune et al., 2005] une chaîne logistique « idéale » présente un haut niveau d'interdépendance entre ses parties, basée principalement sur un partage d'information opportun et transparent afin d'obtenir une congruence entre les objectifs de chaque partie et les objectifs globaux de la chaîne logistique.

La coopération entre entreprises, appuyée par l'usage des nouvelles TICs (Technologies de l'Information et de la Communication), permet théoriquement de gérer le cycle de vie d'un produit continuellement. Dans ce sens, le paradigme PLM (*Product Life Cycle Management*), ou Gestion du Cycle de Vie du Produit, permet de gérer l'information des produits tout au long de leur cycle de vie, de leur conception à leur recyclage [Sudarsan et al., 2005]. L'approche PLM est l'évolution et l'extension du paradigme PDM (*Product Data Management*), lequel est seulement tourné vers la gestion de données techniques du produit [Dankwort et al., 2004]. Le PLM est supporté par une architecture logicielle afin de récupérer, stocker, modifier et gérer l'information des produits et de leurs processus au travers de leurs cycles de vie. Par exemple le projet PROMISE³ (*Product Lifecycle Management and Information Tracking Using Smart Embedded Systems*) a comme vision de développer un système PLM pour permettre à tous les acteurs du cycle de vie du produit de gérer et de contrôler l'information d'un produit *anytime* et *anywhere* [Kiritsis et al., 2003]. Très récemment, la gestion du cycle de vie des produits (PLM) est vue comme un domaine d'application du paradigme intelligence ambiante [Kovács et al., 2006]. Particulièrement, ce paradigme sera expliqué dans le point suivant. Ceci constitue un champ de recherche très riche et prometteur. La problématique de cette thèse est très proche de cette nouvelle interprétation. Dans les points suivants nous présentons les tendances innovatrices qui peuvent être appliquées à la chaîne logistique et à la gestion du cycle de vie d'un produit.

1.2 Nouvelles tendances dans le domaine de la chaîne logistique

Le besoin de disposer des informations associées au produit, qui soient disponible sur le produit lui-même ou sur un autre moyen de stockage, encourage le développement de nouvelles approches favorisant l'association des données aux produits et leur accès en temps réel. Comme exemple de gestion de l'information associée aux produits dans la chaîne logistique, nous citons les services offerts par l'entreprise UPS⁴. Cette entreprise offre des moyens automatiques de gestion de l'information (via Internet, via messagerie électronique,...) afin de suivre les objets physiques et d'informer les clients, en temps réel, des événements importants associés aux paquets à livrer.

³ <http://www.promise.no/> (*Intelligent Manufacturing Systems –IMS / FP 6 – IP*): 2004 – 2008.

⁴ <http://www.ups-scs.fr> - *UPS Supply Chain Solution in France*

1.2.1 Produits et objets communicants

La notion d'objet communicant représente un objet physique capable de communiquer avec son environnement, avec des utilisateurs, d'autres objets ou, en général, avec une entité quelconque qui peut communiquer et interagir avec lui au moyen de technologies de communication filaire ou sans fil [Kintzig et al., 2002]. Le concept d'objet communicant est supporté principalement par le concept de *Ubiquitous Computing*, proposé par Marc Weiser (*Xerox Palo Alto Research*) pour formuler une vision du futur dans lequel les personnes seront entourées d'ordinateurs (de petite taille, mobiles et bon marché) afin d'interagir facilement avec eux d'une façon transparente, invisible et non obstructive (*anytime et anywhere*) [Weiser, 1991]. Dans le Chapitre 2 de ce manuscrit nous allons approfondir ce paradigme. Dans cette optique, l'objet communicant est un objet mobile réagissant à son contexte. La notion de mobilité [Mokhtari et al., 2005] d'un objet qui en est induite est liée à son déplacement physique dans son environnement.

Le consortium « *Thing that think* » au MIT Media Lab⁵ a initié une recherche sur les objets et les environnements augmentés dès l'année 1995 [Borovoy et al., 1996] [Gershenfeld, 1999]. Selon [Ishii et Ullmer, 1997] du MIT Media Lab en attachant des surfaces interactives aux objets physiques, il est possible de lier le monde physique (représenté par des atomes) et le monde virtuel (représenté par des bits). En effet, au travers de l'interaction entre utilisateurs et objets physiques « en touchant les bits » le monde virtuel peut être manipulé, modifié, En fait, les interactions entre un utilisateur et un objet communicant peuvent être explicites ou implicites. Elles sont explicites dans le sens où l'utilisateur manipule directement - il touche - un objet physique afin d'interagir avec sa représentation virtuelle que constituent les bits. De tels objets sont appelés « *graspables objects* » [Van Loenen, 2003]. Dans le cas des interactions implicites, les utilisateurs sont passifs envers l'environnement, en impliquant que les objets ont le droit d'interpréter l'environnement au travers de capteurs, afin de gérer la relation utilisateur / objet / environnement. Dans ce dernier cas, le contexte est clé [Coutaz et al., 2005] pour permettre aux objets d'avoir une vision structurée et unifiée du monde dans lequel ils opèrent. En particulier, les interactions explicites donnent à l'utilisateur l'impression de contrôler son environnement au travers du contact physique avec les objets. En conséquence, le défi est de trouver une balance appropriée entre les interactions explicites et implicites. Dans cet ordre d'idée, [Lahlou et al., 2005] soulignent l'importance d'incorporer au design d'espaces et d'objets communicants les contraintes du caractère privé de l'information. Les auteurs signalent cinq facteurs qui justifient de traiter sérieusement le problème énoncé : la quantité croissante d'objets communicants présents dans notre environnement ; dans un futur très proche, la collection de données dans l'environnement sera pratiquement invisible ; les données seront chaque fois plus personnalisées ; les objets communicants collecteront des données « intentionnellement » afin de nous assister, et finalement, l'existence d'une interconnexion croissante (partage de données) entre des objets communicants.

Une caractéristique des objets communicants est leur capacité à être identifiés automatiquement par les clients intéressés (utilisateurs, processus ou autres objets) dans un environnement de travail donné afin de configurer automatiquement les interactions supportées par les objets communicants. Dans ce sens, les technologies d'identification automatique peuvent être utilisées pour identifier automatiquement un objet communicant. Particulièrement, la technologie RFID [Paret, 2001] [Paret, 2003] [Finkenzeller, 2003] est utilisée pour identifier automatiquement des produits physiques dans la chaîne logistique en permettant, en plus, d'accéder, à distance, à des informations stockées dans des bases de données en utilisant des technologies de communication réseau Internet [Bajic et Chaxel, 2002a]. Le domaine d'application typique des objets communicants est le domaine des

⁵ <http://www.media.mit.edu/> - MIT Media Laboratory

télécommunications [Privat, 2000] [EURESCOM, 2000] [Senn, 2000] [Urnes et al., 2001] [Chioiu et al., 2003]. Dans le domaine informatique, les objets communicants, ou *smart objects*, sont vus comme des objets de la vie quotidienne pouvant accéder à leur image informationnelle disponible dans une infrastructure informatique [Römer et Domnitcheva, 2002] [Mattern, 2003] [Siegemund, 2004]. Comme domaine d'application des objets communicants en pleine effervescence nous citons le domaine de la production et de la chaîne logistique [Wong et al., 2002] [Bajic et Chaxel, 2002a] [Kärkkäinen et al., 2003a] [Cea et al., 2004] [Bajic et Cea, 2005]. Dans ce dernier cas, un objet communicant sera assimilé à un produit intelligent [McFarlane et al., 2002].

L'état de l'art des objets communicants sera présenté dans le chapitre 2 de ce manuscrit. En effet, les concepts présentés dans ce point seront étudiés, approfondis et discutés avec plus de détails.

1.2.2 Intelligence ambiante

Les objets communicants peuvent former des communautés d'objets afin de créer des environnements réactifs et sensibles au contexte. Dans ce cas, les capacités des objets sont mises à disposition dans une situation donnée afin de mettre en œuvre la notion d'intelligence ambiante. Ce concept représente une vision d'un futur très proche où les personnes seront assistées dans leurs activités journalières par des technologies distribuées et noyées dans l'environnement, en prenant en compte activement leur présence et son contexte. Nous avons choisi deux définitions qui résument de façon satisfaisante le concept d'intelligence ambiante ou *Ambient Intelligence*.

“Ambient Intelligence is the vision that technology will become invisible, embedded in our natural surroundings, present whenever we need it, enable by simple and effortless interactions, attuned to all our senses, adaptive to users and context and autonomously acting. High quality information and content must be available to any user, any where, any time, and on any device”. [Lindwer et al., 2003]⁶

“Ambient Intelligence is an exciting new concept in information technology, in which people are empowered through a digital environment that is aware of their presence and context. The environment is sensitive, adaptive and responsive to their needs, habits, gestures and emotions.” [ITEA Ambience, 2004]⁷

Le concept « *Ambient Intelligence* » a été publié la première fois l'année 1999 par [Aarts et Appelo, 1999] et a été adopté officiellement par Philips en 2000⁸. En 2002 Philips a créé HomeLab⁹, un centre de recherche et développement destiné à développer des démonstrateurs du concept d'intelligence ambiante pour évaluer la faisabilité (*feasibility*) et l'usabilité (*usability*) de scénarios dans une vision de maison du futur [Philips HomeLab, 2003]. Selon [Van Loenen, 2003] [ITEA Ambience, 2004] le concept d'intelligence ambiante est le résultat de l'intégration de deux grandes tendances : *ubiquitous computing* et *social user interfaces*. Cette dernière tendance est appuyée, principalement, sur le travail développé par [Reeves et Nass, 1996]. Ces auteurs, dans l'ouvrage « *Media Equation* », argumentent que les communications entre des personnes et des ressources technologiques (ordinateurs, téléviseurs,

⁶ Définition de consensus élaboré par des membres de Philips Research (The Netherlands), Carnegie Mellon University (USA), Eindhoven University of Technology (The Netherlands), European Commission (Belgium), Infineon Technologies (Germany).

⁷ Résultats de l'Ambience Project (*Ambience Intelligence*) 2001 - 2003 supporté par l'ITEA (*Information Technology for European Advancement*).

⁸ <http://www.research.philips.com/profile/history/index.html> - Historical Perspective - Philips Research.

⁹ <http://www.research.philips.com/technologies/misc/homelab/> - Philips Research - Home Lab.

...) se déroulent de la même façon qu'une communication entre des êtres humains. C'est à dire, la personne interagit avec les machines, en général, selon des règles d'interaction interpersonnelle « *face to face* » telles que : la confiance, la similitude, l'attraction et la distance interpersonnelle entre un utilisateur et une machine.

Un autre paradigme supportant des projets de recherche dans une vision d'intelligence ambiante est le paradigme de « *Pervasive Computing* ». Celui-ci a été proposé par IBM à la fin des années quatre-vingt-dix en considérant comme référence fondamentale le paradigme d'*ubiquitous computing* [Ark et Selker, 1999]. En particulier, *pervasive computing* représente l'informatique diffusée dans l'environnement (*everywhere*) afin de faire interagir un utilisateur avec d'autres entités informatiques dans différents scénarios d'interaction. Selon [Mark, 1999] le paradigme *pervasive computing* est associé à une vision du futur dans lequel les ordinateurs font partie de l'environnement et les interactions entre un utilisateur et l'environnement sont dépendants du contexte.

1.3 Les objets communicants dans des projets de recherche

A titre d'exemple nous citons le projet OXYGEN¹⁰ mené au MIT. Ici, la vision du projet est de créer un monde dans lequel la technologie sera centrée sur l'utilisateur et sera disponible partout comme l'oxygène dans l'air que nous respirons. Le projet repose sur trois éléments clés : un dispositif (ou objet communicant) mobile et multi-fonctionnel (PDA, téléphone, caméra, ...) ; des dispositifs (objets communicants) distribués dans l'environnement pour surveiller et contrôler l'environnement et capter la présence d'un utilisateur, sa disponibilité et ses besoins ; et une architecture pour découvrir, localiser et configurer automatiquement les ressources informatiques dans l'environnement [MIT Project Oxygen, 2002].

Le projet « ePerSpace¹¹ » (2004 – 2006) supporté par l'IST (*Information Society Technologies*) et partiellement financé par le programme européen FP 6 (*Sixth Framework Programme*), a eu pour objet la personnalisation des interactions ubiquistes dans un environnement donné (maison, hôtel, aéroport, ...) au moyen de mécanismes pour adapter les services offerts par les dispositifs (objets communicants) en fonction du contexte de l'utilisateur et de ses préférences. En plus, l'accès à ces services peut être effectué à distance (depuis un téléphone mobile, un système de communication dans une voiture, un poste de travail dans un bureau, ...) [Bilchev et al., 2005].

L'initiative européenne « *Disappearing Computer (DC)*¹² » (2001 - 2003), financée par l'IST-FET (*Information Society Technologies - Future and Emerging Technologies*), a été formulée avec le but d'explorer comment la vie quotidienne peut être supportée et améliorée au travers de l'interaction d'objets physiques enrichis avec des technologies de l'information et de la communication [Workshop DC, 2004]. L'initiative a regroupé 17 projets, inspirés dans le paradigme *ubiquitous computing*, afin d'explorer des applications dans des domaines tels que : des vêtements communicants¹³, des objets sensitifs [Rheingold, 2005]¹⁴ (avec de capteurs intégrés), des espaces communicants, des papiers communicants et des tissus communicants.

¹⁰ <http://oxygen.lcs.mit.edu/> - MIT Project Oxygen

¹¹ <http://www.ist-eperspace.org> - ePerSpace Project

¹² <http://www.disappearing-computer.net> - The Disappearing Computer Initiative.

¹³ **Wearable computing** : Concept représentant des ordinateurs intégrés dans les vêtements [Rantanen et al., 2002]

¹⁴ Prestigieuse futurologue américain qu'analyse l'impacte de communiquer des groupes de personnes dispersés géographiquement afin de les faire agir collectivement grâce à des technologies de l'information et la communication.

Le projet AMIGO¹⁵ (*Ambient Intelligence for the networked home environment*), financé par l'IST FP6 (2004-2008), applique le concept des objets communicants au domaine de la domotique. Le but est de proposer une infrastructure technologique ouverte, interopérable, autoconfigurable et dynamique pour faire interagir les dispositifs (objets communicants) d'une maison avec un utilisateur et son contexte. Par rapport à ce projet, dans [Röcker et al., 2005] les besoins exprimés par des utilisateurs, d'origine multiculturelle, sont mises en évidence.

Le projet SIRENA¹⁶ (*Service Infrastructure for Real-Time Embedded Network Applications*), financé par l'ITEA (2003 - 2005), définit une infrastructure ouverte et distribuée pour communiquer des dispositifs (objets communicants) et des applications spécialement dans le domaine industriel (*industrial automation*). Les caractéristiques définies par cette infrastructure sont : l'interopérabilité et la communication directe entre dispositifs (*device to device*), la composition et l'agrégation de dispositifs et la configuration automatique de dispositifs dans l'infrastructure [Jammes et Smit, 2005].

On constate dans ces projets, que la notion d'objet communicant est associée à un dispositif (*device*) informatique mobile ou fixe. Notre intérêt est de permettre qu'un produit physique dans son cycle de vie soit vu comme un objet communicant. L'approche que nous défendons dans cette thèse est de considérer la chaîne logistique comme un environnement d'intelligence ambiante dans lequel les produits supportent des interactions avec leur environnement.

1.4 Problématique scientifique

La problématique de notre recherche est centrée sur la modélisation des interactions entre des produits physiques et des acteurs de la chaîne logistique. Un acteur de la chaîne logistique peut être un processus, un utilisateur ou un autre produit. Nous visualisons un produit physique comme un acteur actif dans son cycle de vie. Ainsi, chaque produit peut être considéré comme susceptible responsable d'organiser les ressources matérielles et informationnelles dans les phases de son cycle de vie afin de gérer son évolution et ses comportements ainsi que ceux de son environnement.

Dans cette approche centrée sur le produit, les interactions entre un produit physique et les acteurs de la chaîne logistique sont déterminées par le profil de ces acteurs et sa situation contextuelle. Le but est de personnaliser les interactions entre un produit et les acteurs de la chaîne logistique au long du cycle de vie d'un produit. La personnalisation, comme paradigme d'interaction, a été traitée et justifiée comme facteur de différenciation pour augmenter la satisfaction des clients [Rousseau et al., 2003] [Console et al., 2003] [Bilchev et al., 2005] [Zemirli et al., 2005]. Particulièrement, dans le domaine de la production, on peut parler d'une personnalisation totale d'un produit (idéale et très coûteuse) ou d'une personnalisation partielle d'un produit (faisable en profitant des économies d'échelle) [Da Silveira et al., 2001].

Nous visualisons un produit physique, chargé d'un rôle actif, comme capable de contrôler ses actions dans les phases de son cycle de vie. Dans ce cas, la réactivité d'un produit face aux changements inattendus dans son cycle de vie doit être considérée. Ceci peut être atteint avec une approche distribuée dans laquelle la structure de communication tend à être horizontale afin d'éviter la rigidité des systèmes hiérarchiques conventionnels [Bongaerts et al., 2000]. Par exemple dans la phase de production, le produit peut avoir un lien permanent avec le client afin d'établir une relation personnalisée produit – client pour permettre à ce dernier de suivre le produit ou intervenir sur sa

¹⁵ <http://www.amigo-project.org> - AMIGO Project.

¹⁶ <http://www.sirena-itea.org> - SIRENA Project.

trajectoire (modifications, compléments,). Dans ce cas, le produit sera le porte-parole du client dans le système de production.

Comme exigence implicite, les entreprises doivent assurer la traçabilité des produit en s'appuyant sur des systèmes qui peuvent assurer une observabilité permanente de chaque instance de produit [Jansen-Vullers et *al.*, 2003]. Dans ce contexte, l'identification individuelle de chaque produit devient un moyen idéal pour identifier et tracer toutes les interactions dans lesquelles un produit a participé dans son cycle de vie.

Dans ce contexte dynamique, la problématique de notre recherche peut être formulée par les questions suivantes :

1. Comment pouvoir faire communiquer un produit physique avec les acteurs de la chaîne logistique à tout moment et dans toute situation ?
2. Comment faire interagir automatiquement un produit physique avec les acteurs de la chaîne logistique au long de son cycle de vie ?
3. Comment gérer l'interaction produit - acteur au long du cycle de vie du produit physique ?

1.5 Structure de la thèse

L'objectif de cette thèse doctorale peut être énoncé de la manière suivante :

« Définir, développer et déployer une approche méthodologique afin de faire interagir les produits physiques et les acteurs de la chaîne logistique dans son cycle de vie permettant de concevoir des architectures technologiques ouvertes pour supporter les interactions produit – processus, à tout moment et en tout endroit, en considérant l'hétérogénéité existante des architectures techniques opérationnelles des systèmes d'entreprises ».

Nos travaux sont basés sur les avancements des domaines des technologies de l'information et de la communication. Notamment la maturation du paradigme *Ubiquitous Computing* permet de développer des nouvelles approches dans le domaine de la chaîne logistique afin de transformer un produit physique en un objet communicant qui est capable d'inférer avec son environnement durant son cycle de vie.

Dans le chapitre 2 nous présentons un état de l'art décrivant la problématique des objets communicants. Comme contribution importante dans cette partie nous soulignons la présentation et l'analyse d'une typologie d'objets communicants. Grâce à cette analyse, nous avons identifié qu'un produit physique peut être représenté comme un demandeur ou un fournisseur de services dans son cycle de vie. Ceci implique, que chaque produit physique peut offrir ou demander des services aux acteurs de la chaîne logistique. Egalement dans ce chapitre, nous remarquons que les objets communicants doivent être identifiés automatiquement dans leur environnement. Particulièrement, la technologie d'identification automatique par radio fréquence (RFID) a été identifiée comme le moyen d'identification d'objets le plus utilisé dans les principaux projets de recherche liés à ce domaine. De plus, à ce jour, cette technologie peut atteindre des niveaux de performance très élevés. Cette technologie permet aussi d'être utilisée par les utilisateurs d'une façon transparente et invisible. Nous terminons ce chapitre avec une description détaillée de cette technologie.

Dans le chapitre 3, nous analysons le concept de « service ambiant » et de « réseau ambiant » afin d'atteindre la mobilité et l'adaptabilité des services d'un objet communicant en fonction de son contexte, d'une façon continue et permanente dans une infrastructure informatique. De plus, nous définissons les fonctionnalités génériques d'une architecture de services ambiants afin de faire interagir un dispositif avec son environnement. Ces fonctionnalités sont : l'identification d'un dispositif, la découverte des services d'un dispositif, la description de services d'un dispositif, l'invocation de services d'un dispositif et la souscription aux services d'un dispositif. La notion d'ambiant fait référence au fait que les services seront disponibles dans un domaine d'interaction donné (*anytime / anywhere*) au moyen d'un réseau de communication. Egalement, nous analysons les architectures de services ambiants pertinentes à ce jour. Comme résultat de cette analyse, nous identifions l'architecture UPnP (*Universal Plug and Play*)¹⁷ pouvant être utilisée comme architecture de référence afin d'enrichir un dispositif avec des services ambiants. A partir de ceci, nous proposons un modèle conceptuel d'un objet communicant en incorporant des fonctionnalités génériques pour permettre une gestion dynamique et automatique de ses services ambiants. La faisabilité de notre modèle d'objet communicant a été validée par des démonstrateurs de laboratoire.

Dans le chapitre 4, nous proposons une méthodologie d'interaction produits / processus permettant de faire interagir automatiquement un produit ou objet communicant avec des acteurs de la chaîne logistique sur la base du concept de service ambiant. Spécifiquement, les acteurs de la chaîne logistique pourront identifier automatiquement un produit, localiser un produit, découvrir les services d'un produit, accéder à la description détaillée des services d'un produit, invoquer les services d'un produit à distance et souscrire aux événements associés à un produit. En effet, dans nos travaux les interactions entre un objet physique et les acteurs de la chaîne logistique sont supportées par les services que chaque acteur met à disposition dans une infrastructure de services ambiants. Dans notre méthodologie un produit peut être un fournisseur de services (rôle passif) ou un demandeur de services (rôle actif). Dans notre proposition, la technologie RFID est utilisée pour identifier automatiquement un objet physique dans un domaine d'interaction. Notre méthodologie supporte les interactions : produit / objet communicant - acteur de la chaîne logistique (processus, utilisateur ou un autre produit / objet communicant). Le concept O2O (*Object to Object*) oriente les interactions entre objets communicants. Finalement, nous proposons des classes de services pour un produit / objet communicant pour son cycle de vie.

Dans le chapitre 5, nous montrons un cas étude de la gestion d'un entrepôt par les services des objets physiques communicants. Ce cas d'étude est une extension de la faisabilité de notre approche méthodologique. Nous avons élaboré des démonstrateurs en considérant des services spécifiques pour les produits ou objets communicants. Effectivement, nous avons pu faire communiquer un objet physique avec des points de contrôle représentant des acteurs de la chaîne logistique, nous avons pu faire interagir ces acteurs avec un objet physique au moyen des services ambiants disponibles dans ce contexte, et finalement nous avons pu tester les interactions en fonction du type d'objet, des profils des acteurs et de la situation particulière d'entreposage de produits.

Les résultats de ces travaux ont été publiés dans : 1 conférence nationale (MOSIM 2004), 1 présentation au GDR MACS (GT OGP), 3 conférences internationales (IFAC Chile 2004, IFAC Prague 2005, ICPR Italy 2005) et 1 séminaire au CDAC (Université de Cambridge, UK). Egalement, des publications internationales (un chapitre de livre et une revue internationale) sont en cours de soumission.

¹⁷ www.upnp.org - UPnP Forum.

2 Etat de l'Art sur les Objets Communicants

2.1 Introduction aux objets communicants

Le concept d'objet communicant apparu au début des années 90 [Borovoy et *al.*, 1996], représente un objet physique capable de percevoir son environnement et d'interagir avec d'autres objets, avec un système d'information et avec des utilisateurs [Beigl et *al.*, 2001]. Plus précisément, un objet communicant est un objet doté de capacités de mémorisation, de perception, de communication, d'action et de décision [Bajic et Chaxel, 2002a]. La capacité de mémorisation est associée à la gestion d'une mémoire locale ou externalisée permettant de stocker des informations caractéristiques de l'objet et essentielles à son cycle de vie. La capacité de perception est associée à la surveillance de l'environnement physique et informatique ainsi qu'aux événements observés dans l'environnement pour s'adapter et gérer son évolution. La capacité de communication est associée à la recherche et la sélection des informations localement ou dans l'environnement dans lequel il est situé. La capacité d'action est associée au traitement de l'information et à l'exécution des mécanismes appropriés à fin d'accomplir une tâche déterminée. La capacité de décision est associée à l'analyse de son état, au comportement des autres objets et au comportement de l'environnement pour prendre des décisions afin de générer des réponses adaptées à des situations particulières. La Figure 2 illustre les capacités décrites d'un objet communicant. Dans la Figure 2 l'objet communicant A interagit avec l'objet communicant B grâce à ses capacités de mémorisation, de perception, de communication, d'action et de décision.

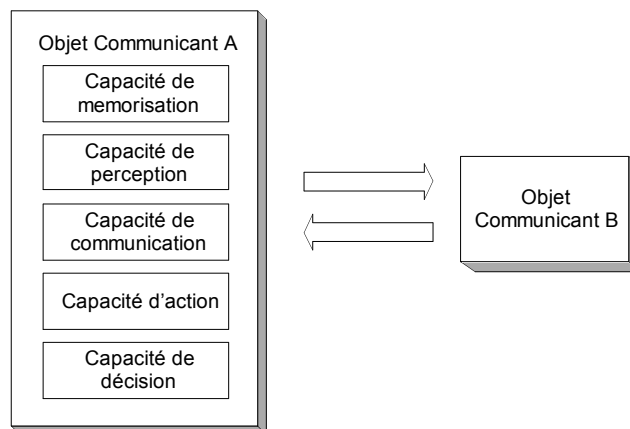


Figure 2 : Les capacités liées à un Objet Communicant.

D'après le livre « Objets communicants » [Kintzig et *al.*, 2002], l'objet communicant est caractérisé comme « un objet physique interagissant, directement ou par le biais de réseaux de communication de nature quelconque, avec son environnement physique, d'autres objets communicants et/ou des utilisateurs humains éventuels, doté au minimum de capacité de mémorisation numérique d'état, et, le cas échéant, de capacité de traitement numérique ». Selon ces auteurs, un objet communicant a les propriétés suivantes : l'enrichissement numérique d'un objet plus que la substitution d'un objet physique ; le caractère transportable des objets et la capacité à communiquer de façon autonome. De plus, une caractéristique important d'objet communicant est la gestion de sa propre source d'énergie [Mattern, 2005].

La Figure 3 illustre quelques exemples d'objets communicants. On observe ainsi dans la figure un téléphone GSM (Nokia¹⁸) et un PDA (*Personal Data Assistant*) équipé d'une antenne RFID Omrom¹⁹ capables d'interagir avec des objets physiques grâce à la technologie radiofréquence, un capteur sans fil Crossbow²⁰ et un micro-contrôleur (μ part)²¹ doté de capteurs intégrés. Typiquement les capteurs sans fil incorporent la notion d'actuateur en utilisant des LEDs (diodes électroluminescente) ou des sons afin d'effectuer des notifications de valeurs critiques aux utilisateurs.



Figure 3 : Exemples de dispositifs Objets Communicants.

Dans les capacités d'un objet communicant, nous pouvons observer l'importance du contexte comme élément de base pour situer les interactions d'un objet communicant avec des éléments externes [Beigl et Gellernen, 2003] [Holmquist et *al.*, 2003]. Le concept de contexte est défini par [Kintzig et *al.*, 2002] comme ce qui n'intervient pas explicitement dans la résolution d'un problème mais contraint cette résolution. Dans cet ordre d'idée, pour un système informatique donné, le contexte représente l'ensemble d'informations maintenu par ce système et indispensable aux traitements des informations échangées avec l'environnement. Un contexte se compose d'une partie statique dont le système possède une description immuable, et d'une partie dynamique qui varie au cours de l'utilisation du système. Une autre définition de contexte très acceptée dans le domaine de l'informatique actuelle a été donnée par [Dey, 2001] : « *Context is any information than can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves* ». Un autre auteur renommé, [Schmidt, 2002] propose d'interpréter la notion de contexte en fonction de la situation, c'est-à-dire, en considérant la localisation, l'entourage ou l'état d'un dispositif comme des éléments implicites pour déterminer le contexte. Un objet communicant peut utiliser l'information contextuelle (situation statique), ou peut s'adapter à un contexte dynamique. Cette notion d'adaptation nous amène à considérer un objet communicant comme une entité utilisant l'information contextuelle pour gérer son évolution. Dans ce sens, nous pouvons souligner qu'un objet communicant peut être caractérisé comme un objet *context-aware* [Dey, 2001] s'il emploie le contexte afin de modifier son comportement et/ou ses relations avec d'autres objets communicants en fonction de la tâche qu'il développe à un instant donné.

¹⁸ www.nokia.com - Nokia

¹⁹ www.omron.com - Omrom

²⁰ www.xbow.com/ - Crossbow

²¹ <http://particle.teco.edu/> - TECO Particle Web Site - University of Karlsruhe

Dans le Tableau 1, nous montrons différents types d'informations contextuelles qui peuvent être associées à un objet communicant. Ce tableau propose des questions qui permettent de caractériser le contexte d'un objet communicant. Cette information contextuelle détermine les interactions que peuvent développer les objets communicants. Dans ce sens, les objets communicants peuvent interagir avec des utilisateurs, d'autres objets communicants, des serveurs d'information [Privat, 2000]. Dans les points suivantes nous allons approfondir la notion d'objet communicant en détaillant les paradigmes concernés (*ubiquitous computing* et *pervasive computing*) et ses dérivations dans le domaine de l'informatique (*smart object*) et des systèmes de production (produit intelligent).

Question	Description de la question
Qui ?	Qui sont les objets qui entourent un objet communicant ?
Combien ?	Combien d'objets sont dans l'environnement de l'objet communicant ?
Quoi ?	Quelles activités sont capables de développer les objets communicants ?
Pour quoi ?	Quelle est la finalité de chaque objet communicant ? Pour quoi sont ils présents dans cet environnement ?
Comment ?	Quel est l'état des objets communicants dans l'environnement ? Comment est caractérisé l'environnement physique des objets communicants ?
Où ?	Où se trouvent physiquement les objets communicants ?
Quand ?	A quel moment les objets sont-ils disponibles dans l'environnement ? A quel moment les objets communicants peuvent-ils interagir avec des autres objets ?

Tableau 1 : Des questions pour identifier l'information contextuelle d'un objet communicant.

2.1.1 Le paradigme *Ubiquitous Computing*

L'expression « *Ubiquitous Computing* » (informatique ubiquiste), a été inventée il y a plus de dix ans par le Dr. Mark Weiser [Weiser, 1991], qui était responsable de recherche au Xerox Palo Alto Research Center (PARC)²². Le programme de recherche « *ubiquitous program* » a été initié à PARC à la fin des années quatre-vingts. Le programme a été envisagé comme une alternative radicale à l'informatisation traditionnelle, basée sur le micro-ordinateur PC, prônant le paradigme puissant à cette période « une personne – un ordinateur de bureau » : trop complexe à employer car il exige trop d'attention ; trop distant des personnes et des activités de la vie quotidienne ; et trop dominant en colonisant des bureaux et nos vies. L'objectif du programme était de redéfinir la relation entre les humains, l'environnement de travail et de la technologie informatique par une ère post ordinateur PC. Dans cette démarche, les ordinateurs sont considérés comme des éléments à repositionner dans l'environnement technologique pour donner lieu à une approche privilégiant l'être humain et ses interactions avec des tâches supportées par des ordinateurs devenant « invisibles » pour les utilisateurs : « *The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it* » [Weiser, 1991]. Les prototypes élaborés chez Xerox (*the board, the pad, the tab, and the badge*) ainsi qu'une infrastructure informatique flexible, ont marqué une nouvelle conception de ce que les ordinateurs pouvaient être [Weiser et al., 1999]. Les prototypes pouvaient reconnaître non seulement le nom d'un dispositif, mais aussi sa localisation, sa situation, son usage, sa connectivité, et son propriétaire. La citation suivante illustre la vision très claire qu'a eue Marc Weiser, auteur du concept, en signalant que les interactions émergeant entre des dispositifs constituent l'élément clé de cette approche : « *Prototypes tabs, and*

²² <http://www.parc.xerox.com/> - Palo Alto Research Center - Xerox

boards are just the beginning of ubiquitous computing. The real power of the concept comes not from any one of these devices; it emerges from the interaction of all of them” [Weiser, 1991].

Dans le domaine des objets communicants [Kintzig et al., 2002] signale que l’ubiquité permet à l’utilisateur d’accéder à des services et de collaborer avec ses collègues à tout instant et où qu’il soit par l’intermédiaire d’outils coopératifs sur supports mobiles. Selon [Weiser et Seely, 1997] l’implémentation du paradigme *ubiquitous computing* nécessite un type de technologie appelée « *calm technology* », décrivant l’état d’esprit désiré des utilisateurs afin de percevoir la technologie comme un moyen confortable pour faire une tâche donnée : “*Just as good, well-balanced hammer “disappears” in the hands of a carpenter and allows him or her to concentrate on the big picture. We hope that computers can participate in a similar magic disappearing act. But it is no simple*” [Weiser et al., 1999].

La Figure 4 montre un exemple d’interactions possibles ubiquistes entre un produit physique, un utilisateur, un camion de distribution, un chariot situé dans un entrepôt, et un endroit de stockage. Dans ce cas, tous les éléments peuvent être connectés à distance par un réseau global de façon permanente afin de communiquer et interagir en fonction des besoins. Les interactions entre ces éléments peuvent à tout moment être développées de manière transparente et non intrusive afin de représenter un scénario d’application du paradigme *ubiquitous computing*.

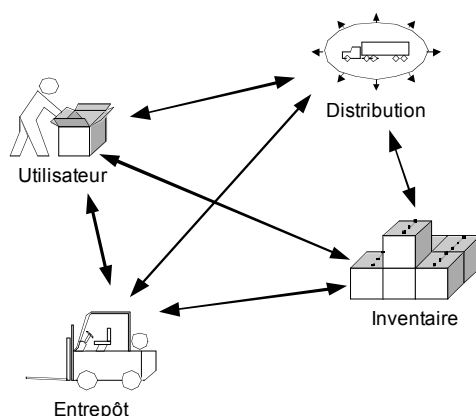


Figure 4 : Exemples d’interactions ubiquistes entre des utilisateurs et des objets.

Selon [Mattern et Sturn, 2003] [Mattern 2005] les tendances et développements qui permettent aujourd’hui d’imaginer un monde informatisé tendant vers l’ubiquité transformant en réalité la vision de Weiser sont :

- des composants microélectroniques toujours plus puissants, plus petits et bon marché ;
- de nouveaux matériels d’interface homme-machine capables d’afficher et de présenter l’information (par exemple une feuille LCD flexible) ;
- des réseaux de communication filaires et sans fil de plus en plus performants et robustes ;
- de multiples capteurs sans fil consommant de moins en moins d’énergie ;
- la possibilité d’identifier de façon distante des objets, et des technologies permettant de localiser ces objets et d’entrer en relation avec eux.

Toutefois, les conséquences sociales, politiques et éthiques de ce paradigme sont encore un domaine complexe à étudier et à analyser de par le caractère futuriste et interdisciplinaire de cette

transformation des relations [Bohn et *al.*, 2005], et principalement en ce qui concerne la protection de la vie privée (CASPIAN²³).

En synthèse, le paradigme *ubiquitous computing* a créé un nouveau champ de l'informatique, en projetant un monde physique enrichi et invisible pour l'utilisateur, entrelacé par des capteurs, des actionneurs, des interfaces et des éléments informatiques insérés « *seamlessly* » dans des dispositifs et connectés entre eux au travers d'un réseau. Ceci nous amène à affirmer que l'invisibilité est l'aspect le plus important dans l'approche *ubiquitous computing*.

2.1.2 Le paradigme *Pervasive Computing*

Malgré que les termes « *ubiquitous computing* » (informatique ubiquiste) et « *pervasive computing* » (informatique diffuse) soient utilisés parfois comme synonymes, il existe des différences conceptuelles entre ces deux approches. Selon [Kintzig et *al.*, 2002] l'informatique diffuse ou « *pervasive computing* » représente un paradigme des technologies de l'information qui repose sur les trois principes suivants : technologie invisible ou transparente pour l'utilisateur, technologie disponible partout et renforcement du couplage des mondes réel et informationnel. Une simple comparaison de cette définition et celle donnée par les mêmes auteurs pour le terme *ubiquitous computing*, nous indique que le paradigme de l'informatique ubiquitaire diffère du paradigme de l'informatique diffuse dans la considération de la mobilité comme variable requise dans le paradigme de l'informatique ubiquitaire. Dans cet ordre d'idées, [Singh et *al.*, 2005] considèrent que le paradigme « *ubiquitous computing* » emploie les avantages de « *mobile computing* » et de « *pervasive computing* » pour présenter un environnement global informatique.

Plus précisément, le paradigme « *mobile computing* » cherche à monter des services et à les rendre disponibles sur les dispositifs mobiles en utilisant une infrastructure sans fil. Le but ici est de réduire la taille des dispositifs de sorte qu'ils puissent être portés n'importe où en permettant d'accéder à des capacités de traitement de l'information par des réseaux à grande vitesse. Mais il faut préciser que « *mobile computing* » a quelques limitations associées au fait qu'il ne considère pas explicitement que les dispositifs peuvent acquérir l'information du contexte pendant les déplacements des utilisateurs et s'ajuster en conséquence à des caractéristiques dynamiques. D'autre part, le paradigme « *pervasive computing* » prétend acquérir des informations sur le contexte d'un environnement de travail et à établir dynamiquement les interactions dépendantes du contexte entre toutes les entités. En conséquence, l'approche « *pervasive computing* » devient effectivement invisible aux utilisateurs humains en leur fournissant des informations utiles en fonction du contexte.

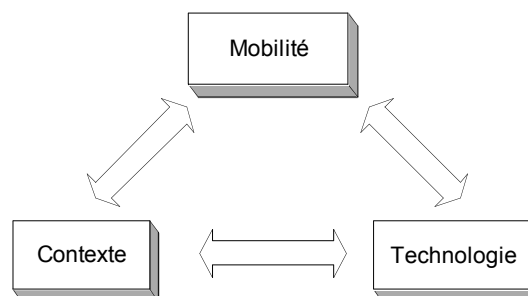


Figure 5: Interactions d'éléments clés selon le paradigme *Ubiquitous Computing*.

²³ C.A.S.P.I.A.N: Consumer against Supermarket Privacy Invasion and Numbering.
(<http://www.nocards.org/AutoID/overview.shtml>)

En résumé, le paradigme « *ubiquitous computing* » a pour but de fournir des environnements informatisés dépendant du contexte de l'utilisateur pendant que celui-ci se déplace d'un endroit à un autre. La Figure 5 illustre la conceptualisation faite par [Yoo et Lyytinen, 2005] par rapport au concept d'*ubiquitous computing*. Selon ces auteurs, l'interaction dynamique, continue et simultanée entre la technologie permettant de faire interagir des ressources (des objets, des applications, ...) afin d'offrir des services à un utilisateur, le contexte de l'utilisateur (description de sa situation physique et sociale) et la mobilité temporelle et spatiale (entre des contextes) d'un utilisateur définit la nature ubiquiste du paradigme *ubiquitous computing*. La complexité de ce scénario d'interaction est augmentée si l'on considère qu'un utilisateur peut avoir plus d'un rôle à la fois pendant son déplacement physique. A l'heure actuelle, l'application du concept « *ubiquitous computing* » est seulement partielle. Nous croyons qu'il faut attendre quelques années encore avant de voir l'application du concept proprement dit dans les différents domaines de la vie actuelle. Compte tenu de ce fait, l'industrie a adopté le concept « *pervasive computing* » pour aller vers une vision technologique invisible centrée sur l'utilisateur [Mattern, 2005].

2.1.3 Le concept de *Smart Object*

Dans l'ouvrage « *When things start to think* » de M. Neil Gershenfeld (*MIT Media Lab*)²⁴ [Gershenfeld, 1999] introduit la notion des « droits des choses », en indiquant trois caractéristiques qu'ils doivent posséder, à savoir : avoir une identité, accéder à d'autres objets et détecter la nature de son environnement. Cette notion est développée dans un rapport stratégique élaboré par quatre entreprises européennes de télécommunications, dont France Télécom [EURESCOM, 2000], en considérant un *smart device* comme un objet physique qui possède un processeur intégré, une certaine mémoire, des capteurs et/ou actionneurs et une connexion réseau. Plus précisément, le rapport mentionne qu'un dispositif doté de telles caractéristiques doit être capable d'agir avec l'environnement et/ou un utilisateur, doit être observable ou contrôlable sur un réseau et être capable de changer son état et ses réponses (sorties) pour fournir un service. L'utilisation d'une interface utilisateur sur l'objet n'est pas obligatoire, elle peut être exportée dans une application distante qui communique avec l'objet au travers d'une connexion réseau [Privat, 2000]. Selon ce dernier auteur, la connectivité réseau entre les objets est l'élément caractéristique.

France Télécom R&D, au travers de l'ouvrage de [Kintzig et al., 2002], définit un objet communicant comme un objet capable d'acquérir, de recevoir et de distribuer de l'information dans un environnement proche ou distant, et de réaliser et/ou de faire réaliser par d'autres objets des actions diverses. Il faut préciser que dans [Kintzig et al., 2003], la version anglaise de l'ouvrage « Objets communicants » parue en 2002, le terme *smart object* est utilisé comme terme équivalent à objet communicant.

Des travaux sur le concept smart objet ont été développés dans l'ETH (*Swiss Federal Institute of Technology*) à Zurich depuis quelques années. Le groupe de recherche (*Distributed Systems Group*)²⁵ conduit par le professeur Friedemann Mattern, créé en 1999, poursuit des recherches dans les domaines de l'informatique répartie et de l'informatique omniprésente (*Ubiquitous Computing*). Pour [Mattern et Sturm, 2003] un *smart object* est un objet capable de s'adapter à l'environnement et de fournir des services utiles en plus de son but original. Il peut être équipé de moyens de connexion aux réseaux spontanés et d'avoir ainsi accès à n'importe quelle information ou permettre d'accéder à n'importe quel service sur Internet.

²⁴ <http://www.media.mit.edu/> - MIT Media Lab

²⁵ <http://www.vs.inf.ethz.ch/> - Distributed Systems Group at the ETH Zurich.

Comme exemple de smart objet nous présentons le prototype *Mediacup* développé à l'Université de Karlsruhe en Allemagne [Beigl et al., 2001]. La *mediacup* est une tasse à café dotée de capacités de perception, de communication et de traitement de l'information afin de gérer l'information contextuelle obtenue à partir de la tasse (Voir Figure 6a). Un scénario d'interaction considère plusieurs *mediacups* en interaction avec d'autres objets tels qu'un écran d'information sur une porte, une montre électronique, une machine à café et un serveur Internet. La Figure 6b montre l'interaction entre une montre et la *mediacup*. Un autre exemple de *smart object* est un *smart-its*. Un *smart-its* est une plateforme (*hardware*) composée de plusieurs types de capteurs, un processeur et une mémoire locale, capable de se communiquer avec d'autres objets par radio fréquence [Beigl et Gellersen, 2003]. La Figure 6c illustre la structure d'un *smart-its*. Un système d'exploitation embarqué sur l'objet permet d'adapter son fonctionnement selon le type d'usage. Le but général est d'attacher un *smart-its* à des objets de la vie quotidienne afin d'augmenter leurs fonctionnalités et transmettre l'information contextuelle dérivée des capteurs.



Figure 6 : Exemples de Smart Objects : l'objet *Mediacup* et le composant intégré *Smart-its*.

Il faut noter que les concepts de *Smart Device* et de *Smart Object* sont le résultat de l'évolution naturelle du domaine de l'électronique embarquée qui a conduit à la décentralisation et à l'enfouissement des capacités de traitement de l'information. Cependant, ces concepts sont essentiellement différents de par la complexité mise en jeu. Un *smart device* est un dispositif communicant complexe de haut niveau, comme par exemple un PDA (*Personal Data Assistant*) ou un téléphone portable. Par contre, la notion de *smart objet* dans nombreux travaux de recherche [Römer et Domnitcheva, 2002] [Mattern, 2003] [Lampe et Flörkemeier, 2004] [Siegemund, 2004] est associée à des objets physiques communicants de la vie quotidienne équipés, au moins, d'un moyen d'indentification automatique pour faire ainsi le lien entre l'objet physique et son image informationnelle disponible dans une infrastructure informatique.

Pour [Mattern, 2003] un objet physique équipé d'une étiquette électronique contenant une adresse Internet comme information numérique peut devenir un *smart object*. Comment ? En utilisant, par exemple, un dispositif mobile (comme un PDA ou un téléphone portable) pour lire cette adresse qui est stockée sur l'étiquette juste en le dirigeant sur l'objet. Ainsi, le dispositif peut afficher l'information associée à l'objet, stocké quelque part, accessible par Internet. Les détails de ce qui est montré dépendront du contexte de l'objet. L'objet communicant avec des capacités de communication et/ou de traitement de l'information peut éventuellement détecter leur environnement par l'intermédiaire de capteurs intégrés [Mattern, 2005]. En ce sens, un objet devient plus autonome et actif dans son cycle de vie. Ils pourraient également communiquer et coopérer avec d'autres objets et, théoriquement, accéder à toute sorte de ressources via Internet. Ceci ajoute une dimension

complètement nouvelle à de tels objets. Par exemple, l'objet pourrait non seulement communiquer avec des personnes et d'autres objets, mais également il pourrait découvrir où ils sont, quels autres objets sont dans leur proximité, et ce qui s'est passé avec eux auparavant (l'histoire des objets).

Selon [Römer et al., 2004], des applications innovantes sont possibles quand les objets réels deviennent *smart object* en ayant de l'information attachée à eux, et puis, étant directement associés à un système informatique ou liés à des services sur l'Internet. Sur la base de quelques expériences pratiques initiales, des prototypes de laboratoire ont été conçus pour tester le concept de *smart object* et son implémentation technologique. La Figure 7 présente le démonstrateur « *smart box* » adapté pour la gestion d'une boîte à pharmacie [Floerkemeier et al., 2003].

En particulier, l'encadré de gauche détaille les capacités nouvelles conférées à l'objet « boîte à pharmacie » afin d'obtenir des informations additionnelles au sujet des objets (par exemple dates d'expiration des objets) ; de connaître l'histoire de l'objet à l'intérieur de la boîte (par exemple depuis combien de temps l'objet a été stocké dans la boîte), d'accéder à distance au contenu actuel de la boîte et d'identifier qui a interagi avec la boîte. De plus, il est possible de surveiller discrètement le contenu de la boîte et de comparer le contenu actuel et le contenu désiré (indication des anomalies). Ce prototype utilise la technologie RFID pour identifier les médicaments.

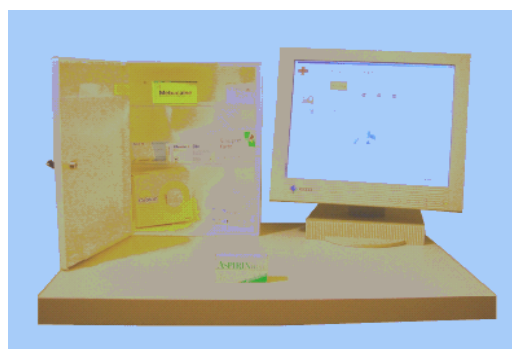
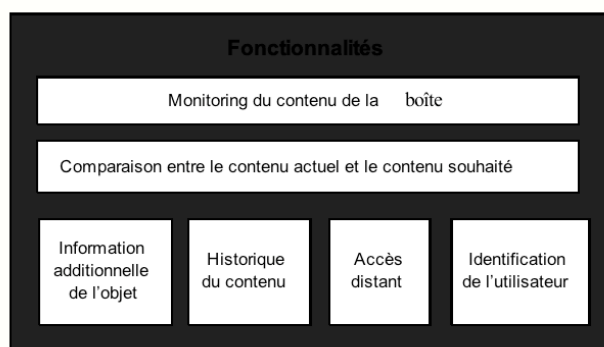


Figure 7 : Le concept de « smart box » et l'application *Smart Medicine Cabinet*.

Comme exemple de *smart object* appliqué au domaine des vêtements, nous citons le concept *smart clothing*, dont le but est d'adapter un vêtement aux changements de l'environnement physique et aux besoins de son utilisateur afin de l'aider dans des scénarios d'interaction prédéfinis [Rantanen et al., 2002]. Ainsi, le vêtement peut agir comme un médium de communication afin d'alerter l'utilisateur dans cas d'une situation particulière (situation d'urgence, ...), il peut agir aussi comme une plateforme offrant de l'information en fonction du contexte (aide professionnelle, aide pour des personnes ciblées, ...) en utilisant une interface intégrée naturellement au vêtement. Comme une application étendue du concept de *smart object*, nous citons le concept d'habitat intelligent pour la santé, qui vise à intégrer des capteurs et des actionneurs pour « suivre » les occupants, communiquer avec eux, et assister intelligemment les occupants dans l'accomplissement des tâches journalières et répétitives [Noury et al., 2003].

En résumé, nous pouvons conclure que le nouveau paradigme d'objets communicants ou *smart objects* s'appuie sur la possibilité d'inclure de nouvelles capacités dans un objet lui permettant de répondre à des demandes d'information faites par des entités intéressées ou de demander de l'information à d'autres entités. Egalement, l'objet peut invoquer des actions complexes, dépendant de sa nature et du contexte, pratiquement n'importe où et à tout moment, grâce à l'utilisation de technologies de

communication Internet, en pouvant ainsi transformer progressivement notre manière de vivre et travailler dans une « société d'objets » nouvellement constituée.

2.1.4 Le concept de Produit Intelligent

Au laboratoire CRAN, la recherche sur le concept « objet communicant » a commencé dans les années quatre-vingt-dix [Bajic et *al.*, 1993]. [Chaxel, 1995] dans sa thèse de doctorat a développé la notion de « objet nomade de production » en exploitant les techniques de l'identification automatique par radiofréquence. Un objet nomade de production est défini comme un objet capable de fournir des données aux clients, coopérer avec les autres entités participant à son évolution, adapter ses règles de décision en fonction de l'expérience acquise durant les phases précédentes de son cycle de production, coordonner des processus de décision et s'auto-finaliser ou imaginer. Dans cette thèse, un système d'information nomade a été proposé, avec les méthodes et les outils associés, où l'initiative du dialogue produit - processus est prise non pas par le produit, serveur de ces informations, mais par les machines, clientes des produits. [Bajic et *al.*, 1999] ont proposé un modèle de référence dans lequel chaque produit et chaque processus est capable : de gérer ses caractéristiques propres (forme, version, ...), de mémoriser la connaissance sur son évolution au travers de la séquence d'opérations d'un produit, de stocker et agréger leur histoire dans les limites technologiques de chaque entité. Dans cette approche, le produit est une entité agissant comme serveur, c'est-à-dire, comme fournisseur d'information et les processus sont des entités agissant comme des demandeurs d'informations vers un produit physique. Dans cette proposition, les produits physiques peuvent stocker leurs informations dans des étiquettes électroniques ou dans des cartes à puce sans contact. Une carte à puce est caractérisée pour permettre le stockage d'informations sur la mémoire de la carte et l'exécution des petites applications (sur un microprocesseur) afin de gérer les opérations réalisées par un utilisateur ou client [Praca et Barral, 2001].

En Angleterre, à l'IfM (*Institute for Manufacturing*), de l'Université de Cambridge, l'équipe de recherche (*Centre for Distributed Automation and Control*) a prolongé la notion de produit intelligent dans le domaine manufacturier. Selon [McFarlane et *al.*, 2002] un produit intelligent est défini comme une représentation physique et informationnelle d'un objet avec les caractéristiques suivantes :

- il possède une identification unique ;
- il est capable de communiquer efficacement avec son environnement ;
- il peut mémoriser ou stocker des données au sujet de lui-même ;
- il dispose d'un langage de communication pour transmettre ses caractéristiques et ses besoins pendant son cycle de vie ;
- il est capable de participer ou de prendre des décisions appropriées à son propre destin de façon continue.

L'identification automatique implique d'obtenir automatiquement l'identité d'un objet au moyen d'une technologie donnée. Par exemple, l'accès en temps réel à l'identité d'un produit grâce à l'identification automatique par radiofréquence permet de déterminer la présence d'un produit et, indirectement il est possible de déterminer sa localisation [Finkenzeller, 2003]. Il est important de signaler que dans la définition donnée de produit intelligent il est possible de distinguer deux niveaux de complexité : un produit apportant de l'information dans son environnement et un produit supportant des mécanismes de décision [Wong et *al.*, 2002]. Ce dernier est le plus complexe parce que dans ce cas il faut doter le produit de mécanismes de décision impliquant que le produit doit avoir une capacité d'analyse intégrée afin d'évaluer et de prendre la meilleure décision en fonction de son état et de son contexte.

Selon [Kärkkäinen et al., 2003a], la notion de produit intelligent est associée à la gestion individuelle de chaque produit au travers de son cycle de vie en intégrant le flux informationnel et matériel (physique) afin d'offrir des services en utilisant un réseau Internet.

Comme exemple concret de l'application du concept de produit intelligent, nous citons la fonction de traçabilité [Jansen-Vullers et al., 2003] de produits dans son cycle de vie. Le but dans ce cas est d'enregistrer et actualiser toute information dynamique associée à un produit (tels que ses états, les opérations qu'il a suivi, ...) sur une étiquette électronique, par exemple, ou sur une ou plusieurs bases de données distantes afin de relier chaque produit physique (individuellement) avec sa représentation informationnelle. Grâce à cela, il est possible qu'un acteur de la chaîne logistique puisse connaître à tout moment l'histoire détaillée d'un produit. Eventuellement, l'information enregistrée sur une étiquette électronique ou sur des bases de données peut être employée comme entrée d'un processus à posteriori afin d'optimiser une opération donnée.

Comme autre exemple, nous citons l'apport de l'information stockée sur un produit intelligent dans un processus de contrôle dans un système de production. La Figure 8 [McFarlane et al., 2003] illustre une boucle de contrôle d'un processus industriel où la capture de données est enrichie avec l'apport de l'information du produit au processus décisionnel au niveau opérationnel grâce à ses capacités de mémorisation et de communication. En effet, l'introduction d'un système d'identification automatique permet au produit physique être reconnu comme telle en apportant ses informations afin d'influer sur les décisions et opérations qu'un système réalise avec lui. Ceci implique d'attribuer un rôle plus actif à un produit physique. Eventuellement la partie décisionnelle associée au produit peut être incluse dans sa représentation virtuelle lui permettant de prendre des décisions autonomes et ainsi devenir une entité active influençant son cycle de vie [Cea et Bajic, 2004] [Bajic et Cea, 2005]. Dans cet ordre d'idées, [McFarlane et al, 2002] affirme qu'un produit intelligent est un article manufacturé qui possède la capacité de surveiller, d'analyser et raisonner au sujet de son état actuel ou futur et, si nécessaire, d'influencer son destin. Comme avantages de cette approche nous pouvons citer la potentialité qu'a un produit de s'adapter (adaptabilité) face aux opérations émergentes (commandes urgentes, pannes des machines, ruptures de stock, reconfiguration des lignes, etc.) ou aux réorganisations de la production, du stockage, ..., et aux besoins dynamiques et potentiellement évolutifs des clients.

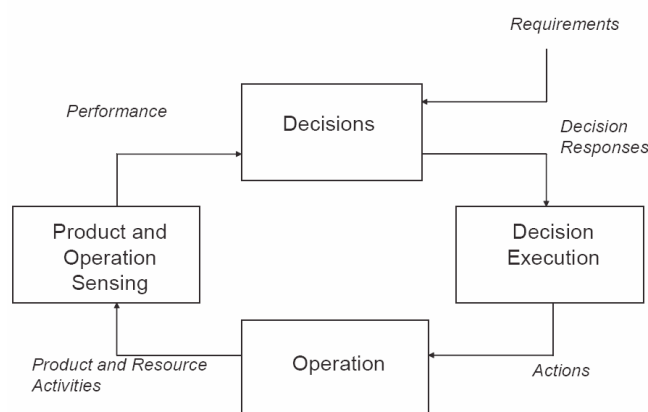


Figure 8 : Apport de l'information du produit dans le processus décisionnelle.

L'introduction de l'identification automatique fournit la possibilité d'employer l'information concernant chaque produit de façon individuelle (son identité, ses états, son histoire, ...) en même temps que les données opérationnelles (tels que la température, position, vitesse, statuts des équipements, ...) afin de contrôler le système. La combinaison de ces deux mécanismes peut

clairement améliorer les processus de décision et de contrôle. Par exemple, la Figure 9 [McFarlane et al., 2003] montre la complémentarité de ces deux mécanismes : l'information stockée sur le produit étant utilisée en même temps que les données opérationnelles afin d'optimiser un processus déterminé. Cette interprétation implique d'avoir une relation « *one-to-one* » entre les éléments de prise de décision et les systèmes physiques effectuant les opérations.

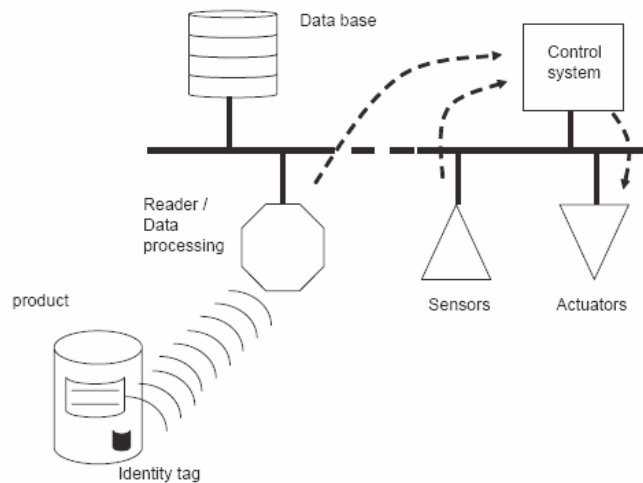


Figure 9 : Optimisation d'un système de contrôle par les informations stockées sur le produit et l'information captée de l'environnement.

La disponibilité de l'information sur le produit et leur processus amène à une visibilité ou observabilité accrue des différents changements des états des produits [McFarlane, 2003]. Ceci, parce que la considération des caractéristiques individuelles d'un produit mène immédiatement à un plus grand nombre d'états possibles. Comme approche complémentaire à l'approche de produit intelligent nous présenterons le paradigme HMS (*Holonic Manufacturing System*).

2.1.4.1 L'influence du paradigme HMS (*Holonic Manufacturing System*)

L'implémentation du concept de produit intelligent dans les systèmes manufacturiers a besoin d'une méthodologie d'ingénierie de systèmes pour modéliser les interactions entre des produits physiques et des ressources matérielles et immatérielles [Morel et al., 2003]. Dans ce sens, le paradigme HMS (*Holonic Manufacturing Systems*) applique le concept d'holon²⁶ aux systèmes manufacturiers. Le concept d'holon a été introduit par Koestler en 1967, il exprime la dualité entre le "tout" et la "partie", c'est-à-dire, qu'une partie d'un ensemble peut former un tout cohérent à lui seul, mais qu'elle est capable de communiquer et de coopérer avec son environnement [Koestler, 1967]. Le projet HMS (*Holonic Manufacturing Systems*) avait pour but de rendre les systèmes de production stables face aux perturbations, de rendre les systèmes de production adaptables et flexibles face aux changements de production, de permettre aux systèmes d'utiliser efficacement les ressources disponibles. Un système manufacturier holonique (HMS) est un système distribué composé par des holons [Valckenaers et al., 1998]. Dans le contexte du projet HMS, un holon est une entité stable, autonome et coopérative. Dans ce but, l'approche holonique pour les systèmes manufacturiers permet de concevoir et d'opérer des modules autonomes et flexibles (holons) afin de former une structure holonique coopérative (appelée *holarchie*). Les fonctions principales d'un holon sont de type opérationnel, sensoriel et décisionnel.

²⁶ Le terme d'holon vient du grec Holos, qui veut dire "tout", et du suffixe « on » utilisé pour désigner une partie ou une particule, comme dans "proton".

L'opérationnalisation de cette approche holonique est basée sur l'approche des systèmes multi agents (SMA) [Ferber, 1995]. Dans ce cas, les agents informatiques peuvent raisonner indépendamment, peuvent réagir aux changements produits par d'autres agents ou son environnement, et peuvent coopérer avec d'autres agents. Ici, le concept clé est la négociation entre agents.

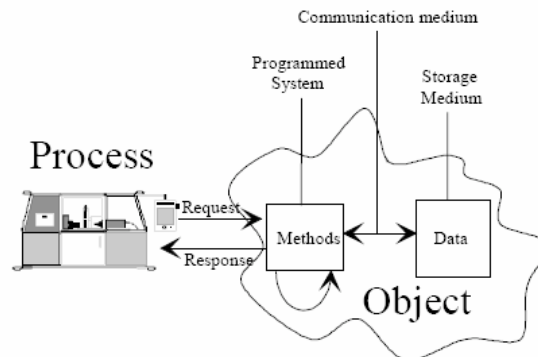


Figure 10 : Approche produit / objet intelligent portant sa propre information.

Inspirés dans ce paradigme, [Bajic et Chaxel, 2002a] ont proposé une approche manufacturière holonique basée sur des produits - holons représentés par une partie du traitement de l'information associée à une partie physique du produit. Dans cette approche, un produit comme entité holonique porte sa propre information sur une étiquette électronique programmable (mode Read/Write) pour communiquer avec chaque client impliqué dans son processus de développement. Les interactions entre un produit et un processus sont développées au moyen de requêtes / réponses comme il est montré dans la Figure 10. Les principales issues de cette approche sont associées à la gestion de l'information, la mobilité du produit et la possibilité de prise de décision de chaque holon-produit. Le produit est le manager de sa propre information et la transmet au processus après réception d'une requête. L'information du produit est classée en deux catégories : l'information sur le produit et l'information distribuée sur une base de données accessible par réseau. Une vision similaire est proposée par [McFarlane et al., 2003] en considérant un produit intelligent comme un holon-produit en intégrant une partie physique, une partie informationnelle et une partie décisionnelle capable de communiquer et de coopérer avec d'autres holons.

2.1.5 Synthèse des concepts

Comme nous avons pu le constater, les concepts de « *smart object* » et produit intelligent sont conceptuellement très proches. Dans les deux cas il est nécessaire d'identifier automatiquement un produit / objet, de disposer d'un langage de communication approprié entre le produit / objet et des ressources informatiques distribuées, afin d'étendre, par exemple, la capacité de stockage disponible sur le produit / objet. La notion d'objet physique et d'objet virtuel est implicite dans les deux approches. Egalement dans les deux approches, les interactions entre des produits / objets et des acteurs d'un système (utilisateurs, machines, processus, autres objets, etc.) requièrent l'implémentation de mécanismes d'action et décision avancés distribués sur le produit / objet et sur une infrastructure informatique pour gérer effectivement la relation produit / objet - contexte dans son cycle de vie. Pour ces raisons, dans ce travail le concept de *smart objet* et *produit intelligent* sont considérés comme des synonymes.

2.2 Classification des objets communicants

Compte tenu qu'il existe une diversité d'implémentation des capacités d'un objet pour atteindre les caractéristiques étendues d'un objet communicant, nous proposons une classification [Bajic, 2004] en considérant quatre types génériques depuis le plus élémentaire jusqu'au plus complexe :

- L'objet porteur de données
- L'objet pointeur vers un système d'information
- L'objet fournisseur et demandeur de services
- L'objet sensitif enrichi avec des capteurs et des actionneurs.

Dans la deuxième partie de ce chapitre nous expliquerons les types d'objet communicant, qui en même temps, peuvent être interprétés comme approches particulières pour conceptualiser un objet communicant.

2.2.1 L'objet porteur de données

L'objet communicant agit comme un porteur ou transporteur de données (bits, nombres, texte, ...). Le but est de stocker dans l'objet les données minimales élémentaires qui seront accessibles par des ressources externes. Ces données sont utilisées dans des points de contrôle ou processus (automate programmable, robot, ...) comme information de multiplexage ou paramétrisation des opérations et transformations effectuées sur l'objet [Jansen et Krabs, 1999]. Il faut préciser que quand une donnée est associée à une sémantique, apparaît alors le concept d'information [Kintzig et al., 2002], la sémantique permettant de donner une interprétation de la donnée (par exemple un numéro avec une unité de mesure).

Pour implémenter cette approche, la technologie de code à barre et la technologie de radiofréquence sont souvent employées comme moyen de transport d'information par identification automatique, principalement dans le domaine de la production industrielle. En conséquent, l'extraction des données d'un objet est réalisée normalement au moyen d'un dispositif de lecture externe. Les formats des données peuvent être variés et peuvent utiliser de multiples codes propriétaires (numérique, alphanumérique, hexadécimal, binaire, ...) ou employer des codes standardisés type code à barre EAN13 par exemple. Nous signalons que l'identification individuelle de chaque produit n'est pas obligatoire, mais s'il est nécessaire, en fonction de la nature de l'application, un code d'identification unique doit être considéré afin d'identifier un objet physique.

Comme moyen de marquage d'objets, il est possible d'utiliser le code à barre unidimensionnel ou linéaire, le code à barre linéaire empilé, ou le code à barre à deux dimensions [Wartenberg et Snyder, 2003]. Le code à barre unidimensionnel est composé d'une série de barres sombres parallèles et des espaces clairs permettant de représenter une quantité fixe ou variable très limitée de caractères numériques ou alphanumériques. Le code à barre linéaire empilé, de longueur variable, utilise des files empilées de barres pour coder les données. Comme exemple de ce type de code, nous citons le code PDF417 (Portable data file standard) incorporant jusqu'à 1.850 caractères alphanumériques. Le code à barres à deux dimensions, de longueur variable, permet d'atteindre une densité (exprimée en caractères / cm²) plus importante par rapport au code à barre linéaire empilé. Comme exemple de code bidimensionnel nous citons le code matriciel incorporant jusqu'à 2.335 caractères alphanumériques. La Figure 11 illustre, à gauche le code PDF417 décrivant la chaîne de caractères « abcd123 », et à droite

la même chaîne de caractères en format matriciel. On observe que le code matriciel est plus dense que le code PDF417.



Figure 11 : Exemple de codes à barre à deux dimensions.

Il est tout à fait possible d'intégrer des données biométriques sur un code à barre. Par exemple, [Noore et al., 2004] ont proposé une approche pour inclure dans un document (permis de conduire) un code à barre PDF417 en intégrant de manière codifiée une image du visage non comprimée du conducteur, une image de l'empreinte digitale non comprimée du conducteur, et quelques données personnelles (prénom, nom, ...). Dans ce cas, un permis de conduire représente un objet porteur de données de type personnel.

Dans le domaine de la chaîne logistique, le code à barre est le moyen d'identification automatique le plus utilisé pour identifier et caractériser un type de produit déterminé et ses avantages sont indiscutables, ils sont peu coûteux et la distance de lecture est raisonnablement bonne. Pour cela, le code EAN-13, géré par l'EAN International, est utilisé afin d'identifier le pays d'origine du produit, le fabricant d'un produit ainsi qu'un numéro d'identification pour le type de produit²⁷. Dans ce cas, en regardant le code EAN-13 sans connaître sa signification ou son décodage, nous pouvons dire que ces codes sont simplement des données utiles pour la traçabilité des produits. Bien sûr l'interprétation du code permet d'avoir l'information nécessaire pour réagir en cas de problèmes. Le code EAN-128 est utilisé dans le cas d'une unité logistique type carton homogène ou une unité équivalente. Pour identifier des unités d'expédition de façon individuelle le code SSCC (numéro séquentiel de colis) est utilisé. La Figure 12 illustre les codes à barre unidimensionnels les plus utilisés dans la chaîne logistique : le code EAN-13, le code EAN-128 et le code SSCC. En résumé, il existe plusieurs niveaux de granularités pour identifier un objet physique.

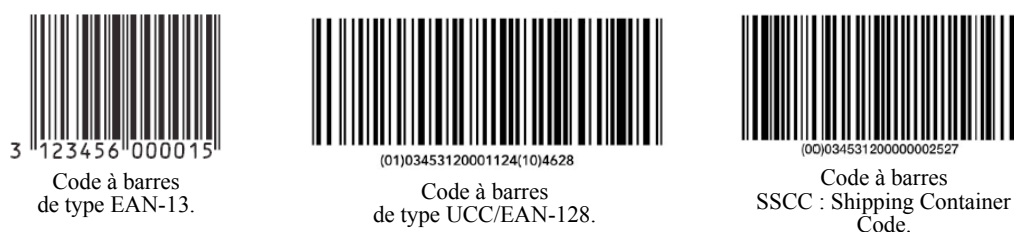


Figure 12 : Des formats classiques de codes à barre linéaires employés dans la chaîne logistique.

Le principal inconvénient de la technologie de codes à barres est associé à son usage, les codes peuvent devenir illisibles une fois soumis à des environnements de manipulation difficile. De plus, une fois que les données sont enregistrées sur un papier ou directement sur le produit, elles ne peuvent plus être modifiées. Nous constatons que la principale faiblesse de la technologie code barre est que les codes ne donnent pas la possibilité d'identifier un produit de façon individuelle. Les étiquettes électroniques peuvent résoudre ce problème en stockant un code d'identification unique pour identifier chaque produit. En plus, sur l'étiquette électronique d'un produit physique, on peut stocker quelques

²⁷ www.eannet-france.org - EAN France

données additionnelles. Un simple numéro indiquant un attribut spécial (la couleur, la forme, la priorité, ...) dans un système de production peut être une donnée importante pour prendre une bonne décision. Dans ce sens, [Finkenzeller, 2003] considère qu'un objet porteur de données, faisant partie d'un système de contrôle centralisé ou décentralisé, permet de synchroniser le flux physique et le flux informationnel des objets. En effet, les états des objets peuvent être suivis grâce aux moyens d'identification automatique.

Ainsi, les capacités étendues d'information de l'objet sont limitées parce que la configuration de l'objet ne considère pas explicitement un lien entre chaque objet individuel et sa représentation virtuelle dans un système d'information.

2.2.2 L'objet pointeur vers un système d'information

Dans cette approche, la fonctionnalité essentielle d'un objet communicant est de fournir une identification unique à une infrastructure informatique afin d'adresser un serveur d'information dans une infrastructure informatique [Bajic et Chaxel, 2002b] [McFarlane, 2002] [Matern, 2003] [Karkkainen et al., 2003a]. Cette infrastructure effectue des opérations au nom des objets en communiquant avec les contreparties virtuelles des objets physiques. En effet, cette approche permet de lier les objets physiques avec une représentation virtuelle dans un monde informatique diffus. L'objet en plus de communiquer son identifiant, éventuellement, peut stocker des données complémentaires grâce à l'utilisation d'une certaine technologie [Cea et al., 2004]. Dans ce cas, l'objet communicant ne possède pas les ressources informatiques nécessaires pour agir d'une façon autonome directement sur d'autres entités (utilisateurs, processus, objets, ...) [Siegemund, 2004].

Un simple lien entre le code à barre d'un produit et un système d'information peut donner accès à des informations utiles pour un certain type de produit. Cette notion est exploitée dans le système proposé par [Saar et al., 2004] en faisant le lien entre un type de produit identifié par son code à barres, et les informations de recyclage personnalisées qui sont accessibles via une page web, comme le montre la Figure 13, pour un objet téléphone portable doté d'un code à barres.

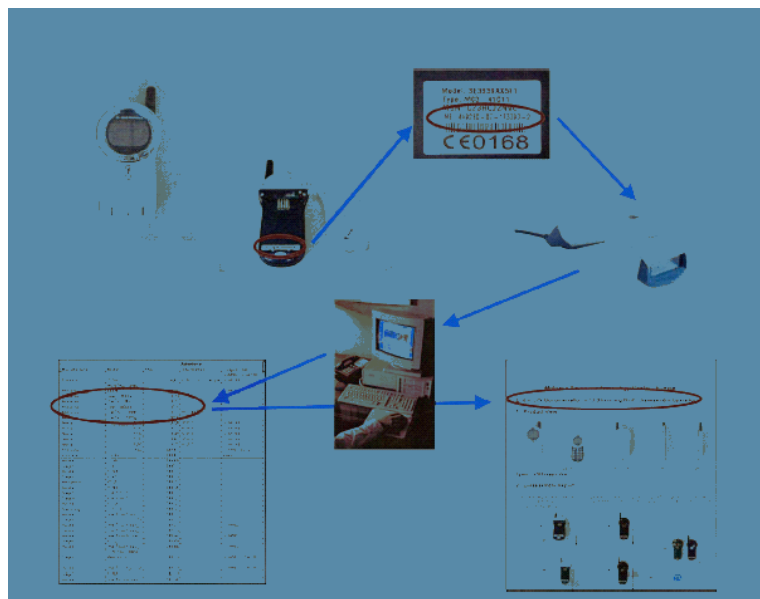


Figure 13 : L'objet pointe vers ses informations de recyclage accessibles par une page Web.

Ainsi, en utilisant un lecteur code barre, une application logicielle peut consulter à distance une base de données par association du code à barre au modèle spécifique du produit.

Dans la Figure 14, l'objet communicant est représenté par l'objet physique A et sa représentation virtuelle se trouve sur une étiquette électronique et sur une base de données distante. L'objet physique A est identifié par le système RFID géré par la Machine A permettant de relier son identifiant avec sa représentation informationnelle sur une base de données accessible par Internet.

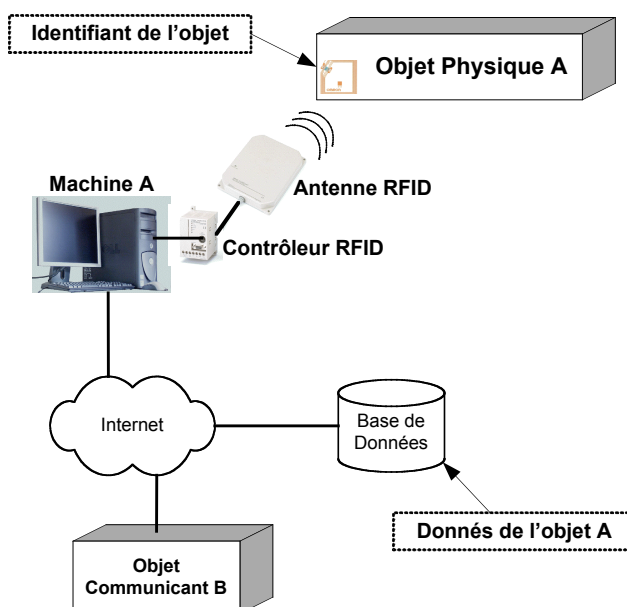


Figure 14 : Approche objet comme vecteur d'information

Comme inconvénient de cette approche, nous pouvons avancer que l'implémentation dépend de la disponibilité des dispositifs externes pour établir le lien entre les objets et leurs représentations virtuelles. De plus, nous pouvons déduire que dans ce cas, il n'y a pas de collaboration directe entre les objets physiques, mais plutôt entre leurs représentations virtuelles. Des informations plus détaillées sur un objet, telles que sa localisation exacte ou sa température, peuvent être inférées à partir de l'endroit où l'identification de l'objet est réalisée, mais il est évident que ceci peut mener à des inexactitudes. La dérivation des informations contextuelles plus sophistiquées sur un objet, tel que son état actuel, exigent souvent des capteurs sur l'objet lui-même.

Nous présenterons ensuite l'application de cette approche selon plusieurs variantes illustrées par des projets de recherche de la communauté scientifique.

2.2.2.1 L'initiative EPC Global

L'initiative EPC Global²⁸, supportée par l'EAN International et l'Uniform Code Council Inc, a permis d'industrialiser le concept développé par l'Auto-ID Lab du MIT (*Massachusetts Institute of Technology*)²⁹. Cette approche propose d'identifier individuellement chaque produit dans son cycle de vie en utilisant un code appelé Electronic Product Code (EPC) [Brock, 2001]. Ce code est un code standardisé de 96 bits (12 octets) qui identifie dans sa composition la version du code EPC (8 bits), le

²⁸ <http://www.epcglobalinc.org/>

²⁹ <http://autoid.mit.edu/cs/>

fabricant de l'objet (28 bits), la classe de l'objet (24 bits) et le numéro de série de l'objet (36 bits) [Engels, 2003]. Le code EPC est représenté dans la Figure 15.

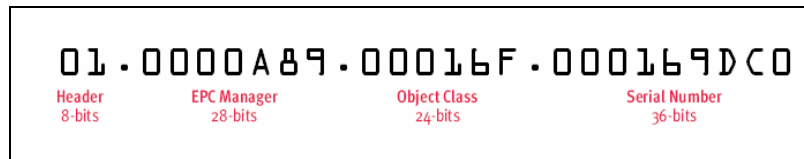


Figure 15 : Code EPC de 96 bits.

L'information associée au produit n'est pas stockée directement dans le code, le code EPC sert de référence à l'information gérée sur le réseau Internet. Autrement dit, le code est une adresse utilisée pour trouver l'information ou l'image virtuelle de l'objet sur l'Internet. La technologie RFID sert de support pour mémoriser le code EPC sur une étiquette électronique. Elle peut être placée dans le produit lui-même, dans un emballage.... Les fréquences RFID choisies pour mémoriser le code EPC sont dans les bandes HF et l'UHF, plus particulièrement la fréquence 915 Mhz, et la fréquence 13.56 MHz. Les composants de base de cette architecture d'information sont : le PML (*Product Markup Language*), le logiciel SAVANT et l'ONS (*Object Naming Service*). Le PML est un langage, basé sur la syntaxe XML, pour décrire les objets physiques [Brock et al., 2001]. Le langage PML décrit les objets physiques, leur configuration et leurs états. Un logiciel SAVANT permet de localiser l'objet virtuel sur Internet (décrit en PML) après l'identification d'un code EPC porté par un objet physique. Un mécanisme de recherche d'information sur Internet appelé ONS, indique où localiser l'information sur l'Internet au sujet de n'importe quel objet qui porte un EPC [Oat Systems et MIT Auto-ID Center, 2002]. L'ONS est semblable au *Domain Name System* (DNS) existant de l'Internet, qui permet à un ordinateur d'identifier où sont stockées les pages liées à un site Web particulier. L'ONS sera employé chaque fois qu'il est nécessaire d'associer l'objet physique à sa représentation virtuelle.

La Figure 16 montre la séquence nécessaire d'opérations à exécuter depuis l'identification du code EPC par un système RFID jusqu'au contact du serveur contenant les informations associées à l'objet identifié. On observe que l'objet répond à une interrogation du lecteur RFID en envoyant le code EPC à un serveur local. Celui-ci est chargé de contacter le serveur d'informations ONS afin de rechercher l'adresse logique du serveur web PML. Lorsque le serveur local a reçu l'adresse IP demandée, il pourra interagir directement avec l'objet physique.

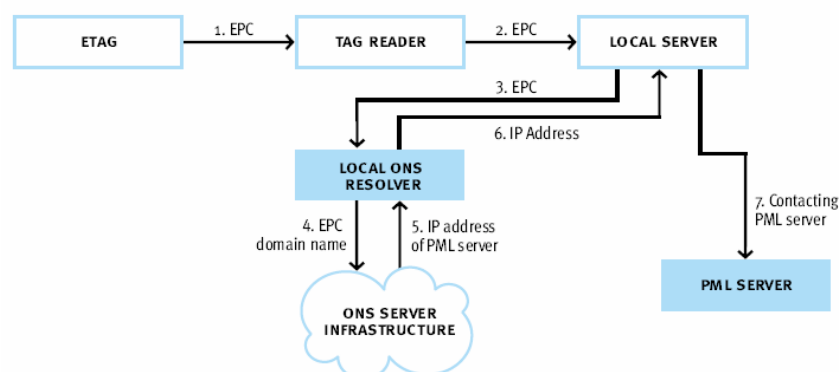


Figure 16 : Architecture EPC d'association de système d'information au Produit.

Comme idée-force de cette approche nous pouvons citer l'impact mondial induit par l'alliance EAN/UCC pour pousser son adoption dans la grande distribution et l'effort de standardisation,

reconnu par l'ISO (Norme ISO 18000-6 version C), qui permet de résoudre le problème associé à l'interopérabilité entre solutions propriétaires. Comme faiblesses potentielles de cette approche, nous pouvons citer la difficulté à avoir une rapide acceptation globale du codage des produits dans le domaine de la chaîne logistique, la gestion centralisée dans l'attribution des codes EPC et la dépendance réseau du système obligeant la consultation de l'ONS avant d'obtenir toutes informations associées aux produits en impliquant une augmentation du trafic réseau et un retard dans l'accès à l'information.

En résumé, cette approche permet de connecter un objet physique avec son image informationnelle via une communication Internet, grâce à l'utilisation d'une clé d'accès spécifique statique (code EPC) obtenue, par un système RFID, après l'interrogation de l'étiquette électronique portée par l'objet.

2.2.2.2 L'objet Internet

Des travaux de recherche menés à l'HUT (*Helsinki University of Technology*)³⁰ ont proposé une autre technologie d'information pour gérer des produits intelligents, appelée « *Dialog System* » (*Distributed Informations Architectures for Collaborative LOGistics*)³¹. Cette approche vise à résoudre les défis de l'identification unique des produits sans besoin de développer de nouvelles normes de codage [Karkkainen et al., 2003a]. Dans cette proposition, le lien entre un objet réel et son adresse Internet, qui permet accéder à l'information concernant l'objet, est défini par deux éléments : la partie d'identification d'un produit physique (un numéro de série, un code propriétaire, ...) et l'URI (*Uniform Resource Identifier*), qui représente l'adresse Internet d'un agent distant associé à l'objet réel (par exemple l'URI pour le fabricant du produit, l'entreprise de transport, ...).

De cette façon, le projet propose d'employer une convention de codage semblable aux adresses e-mail, c'est-à-dire, un format : identifiant@URI (par exemple 12345@dialog.hut.fi). S'il existe une relation biunivoque entre l'identifiant de l'objet et l'adresse de l'entreprise (URI) la combinaison résultante est globalement unique. L'ensemble ID@URI peut être imprimé sur le produit comme texte ou code à barre, ou bien, être programmé dans une étiquette RFID. Le projet emploie des étiquettes I-Code communicant à 13.56 MHz, accessibles en lecture écriture, où le numéro de série de l'étiquette (unique) représente l'identifiant statique de l'objet et l'URI de l'agent distant est stocké dans la mémoire inscriptible de l'étiquette [Karkkainen et al., 2003b]. La Figure 17 représente l'interaction directe entre une application RFID et le serveur d'information associé au produit. En effet, le PDA obtient l'identification unique de l'objet (par RFID par exemple) afin d'obtenir l'information significative associée au produit physique.

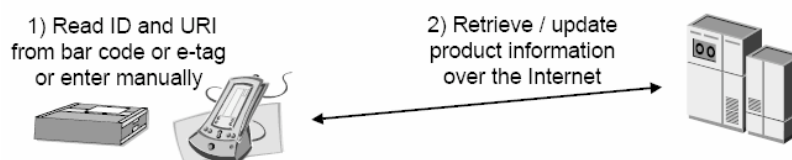


Figure 17 : Le système Dialog.

L'application « *Dialog System* », programmée en Java, fournit des interfaces pour : réceptionner des mises à jour de la localisation des produits, lier l'identifiant avec des références internes d'une entreprise, rechercher et montrer l'information relative d'un produit.

³⁰ <http://www.tkk.fi/English/> - Helsinki University of Technology

³¹ <http://dialog.hut.fi/> - Distributed Informations Architectures for Collaborative LOGistics

Comme avantage de cette approche, nous citons la simplicité du code d'identification unique proposé et l'interaction directe entre le dispositif, qui gère la capacité de communication de l'objet physique, et sa représentation virtuelle. Comme inconvénient de cette approche, nous citons la faible standardisation des identifiants des objets.

En synthèse, l'implémentation technologique de ces deux dernières approches (l'EPC Global et l'Objet Internet), dans lequel l'objet est un pointeur vers un système d'information, requiert l'intégration d'un système d'identification automatique avec une infrastructure informatique opérationnelle à tout moment et accessible n'importe où. En considérant le principe de « *calm technology* » proposé par [Weiser et Brown., 1997], l'utilisation des étiquettes électroniques facilite l'identification des objets d'une façon discrète. De plus, les étiquettes passives fournissant l'identification de l'objet n'exigent aucune batterie, puisqu'elles sont alimentées par des dispositifs externes (lecteurs RFID). Dans ce cas, l'approvisionnement en énergie ne constitue pas un problème. Alternativement l'approche peut être implémentée avec d'autres technologies : lecteurs optiques, lecteurs de signatures, etc.

2.2.2.3 L'objet Augmenté

Le concept de « Réalité Augmentée » appliqué au domaine des objets communicants permet de concevoir des objets dits « augmentés » [Mattern, 2005]. Un objet augmenté est un objet familier, amplifié avec des capacités de traitement de l'information, ou amplifié par des informations numériques superposées et définies par un système informatique [Kintzig et al., 2002]. Afin d'approfondir la finalité de ce type d'objet communicant, nous citons [Falk et al., 1999] qui a contrasté le concept de Réalité Amplifiée avec le concept de Réalité Augmentée. Pour ces auteurs, le concept de réalité amplifiée est associé à la diffusion publique des propriétés d'un objet physique en utilisant des ressources informatiques embarquées dans l'objet. En revanche, la réalité augmentée est associée à une réalité superposée sur un objet qui est de nature privée, c'est-à-dire, les représentations des objets sont présentées à l'utilisateur en fonction de son profil. Dans ce sens, nous pouvons dire qu'un objet amplifié est un objet qui s'exprime tel qu'il est dans un domaine public en incorporant dans sa structure toutes les propriétés qui le décrivent. Un objet augmenté est un objet qui intègre une réalité superposée (tel un objet virtuel dual) sur lui-même et qui est aperçu (subjectivement) comme tel en fonction des droits de l'utilisateur qui interagit avec lui. En effet, l'information augmentée de l'objet permet d'enrichir le monde réel (objet physique) avec des éléments d'un monde virtuel (objet virtuel) en fonction du profil d'un utilisateur. Le but est de fournir de l'information additionnelle aux acteurs de sorte que le produit physique soit enrichi.

Dans cette approche, l'objet en plus d'avoir un identifiant stocké localement (pointeur), dispose d'autres données enregistrées localement sur un support déterminé pour enrichir l'accès à l'information disponible [Bajic et Chaxel, 2002b]. Dans cette approche, une caractéristique importante est d'offrir aux utilisateurs de l'information minimale sans accéder forcément à une infrastructure réseau tierce, c'est-à-dire, qu'une entité peut s'informer localement de certaines caractéristiques de l'objet, sans être connecté sur un réseau informatique. L'interaction d'une entité physique ou informatique avec l'objet peut éventuellement entraîner un changement de ces caractéristiques (mise à jour des données sur l'objet).

Nous pensons que la technologie RFID permet d'implémenter cette approche en permettant d'intégrer les fonctionnalités requises, notamment la lecture et l'écriture des données sur une étiquette RFID attachée à l'objet. Ainsi, sur cette étiquette il est possible de stocker, en plus d'un identifiant numérique, des données additionnelles sur l'objet en fonction de sa capacité de stockage. Un

ordinateur ou un PDA équipé avec un lecteur RFID peut interroger une étiquette afin de connaître l'information minimale à condition de connaître, à priori, la structure des contenus présents dans sa mémoire. En plus, des lecteurs d'étiquette RFID peuvent être fixés dans d'autres ressources du monde physique (sur des machines, dispositifs industriels...).

Afin de démontrer l'application de cette approche nous avons développé le démonstrateur « poster intelligent » [Cea et *al.*, 2004]. Le concept de « poster classique » est associé à une information statique limitée seulement à la nature intrinsèque de l'objet : sa surface. Par contre, un poster intelligent est défini comme un « poster augmenté » permettant d'accéder à son image virtuelle pour avoir plus d'informations de nature extrinsèque mais intégrées à l'objet, de rentrer en relation avec ses concepteurs, voire de proposer des modifications ou extensions sur son contenu. La Figure 18 illustre l'architecture développée : un PDA (*Personal Data Assistant*), équipé d'une antenne RFID et d'une liaison réseau sans fil WiFi, réalise l'interface informatique avec le poster intelligent portant une étiquette électronique 13.56 MHz. La base de données distribuée et accessible par réseau Internet, représente l'extension virtuelle du poster, ou système d'information augmenté. L'accès à l'information est déterminé par le profil de l'utilisateur. Ce démonstrateur a mis en évidence une augmentation des caractéristiques intrinsèques d'un poster, par une information vivante et dynamique, dans un environnement d'interaction sans contact restreint par les profils des acteurs.

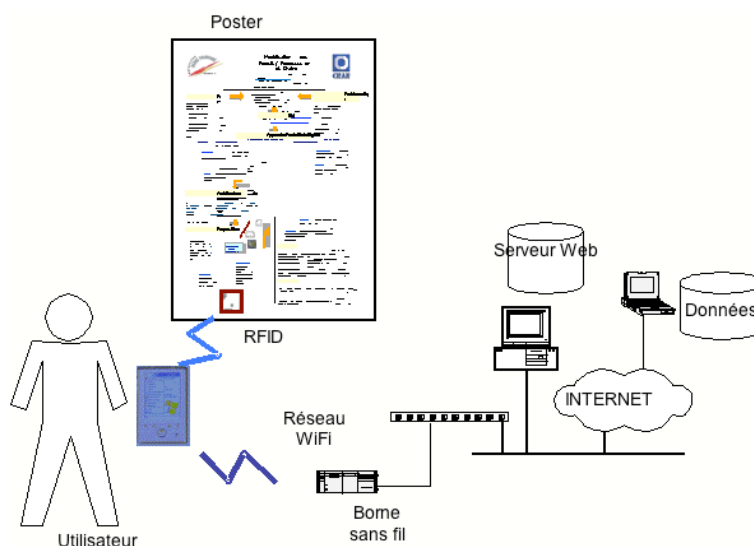


Figure 18 : Utilisateur obtenant de l'information augmentée d'un poster intelligent.

En résumé, cette approche permet d'augmenter les informations intrinsèques d'un objet physique par un lien avec sa représentation virtuelle, projetant des informations augmentées de l'objet comme une réalité augmentée disponible sur une interface de présentation, et permettant de maximiser l'usage de l'objet en fonction du profil de l'utilisateur dans un domaine de travail donné.

2.2.3 L'objet comme prestataire et/ou demandeur de services

Selon [O'Sullivan et *al.*, 2002] un service est une action qui est demandée par une entité au nom d'une autre entité avec le but de transporter une valeur intrinsèque depuis le fournisseur du service au destinataire du service. En outre, ils ajoutent qu'un service peut être contenu dans d'autres services. Dans [Figure, 2004] un service est présenté comme une fonctionnalité qui est offerte à un utilisateur en fonction de sa situation contextuelle, laquelle est déterminée par l'interaction des trois éléments suivants : son profil (attributs, préférences, ...), sa localisation géographique et sa dimension

temporelle (la date, l'heure, des activités planifiées, ...). Dans ce cas, on parle de services dépendant de la situation du demandeur de services. Une notion dynamique du concept de service est représentée par [Gu et al., 2005]. Ici, un service associé à un utilisateur utilise l'information contextuelle pour s'adapter automatiquement aux changements dynamiques de l'environnement en exécutant des actions en fonction de la situation. Dans [Console et al., 2003] les services sont présentés comme fonctionnalités personnalisables selon le profil de l'utilisateur et adaptables à une situation donnée au moment donné (notion d'opportunité dans le service). Concernant le moyen technique utilisé pour accéder à un service nous citons, par exemple, les services web, les services mobiles... Pour [Wang et al., 2004] un service web exécute des fonctions (via une application web) qui peuvent aller d'une simple requête d'information jusqu'à l'exécution à distance de processus complexes. Dans [Ciminiera et al., 2004] les services mobiles sont des fonctionnalités personnalisables en considérant la mobilité de l'utilisateur (*smart card*) et la mobilité du terminal d'accès (PDA). Pour [Georgakopoulos et al., 2002] les entreprises virtuelles sont des organisations fournissant des e-services, ensemble des fonctionnalités coordonnées électroniquement afin de fournir des produits ou des services entre entreprises.

Dans notre approche, le concept de service représente une fonctionnalité, initiée par un objet communicant ou par un acteur de la chaîne logistique, permettant d'invoquer l'exécution d'une ou plusieurs actions à distance afin d'obtenir une réponse apportant des éléments de décision à l'entité qui initie la communication. Ceci implique de pouvoir caractériser un objet communicant comme un objet dual, qui peut être soit un fournisseur de services soit un demandeur de services [Cea et al., 2004]. Etant un fournisseur de services, l'objet offre ses services aux acteurs intéressés selon un modèle d'interactions produit - processus. Etant un demandeur de services, l'objet fait des demandes de services aux acteurs qui ont des services disponibles. Il est important de noter que l'objet étant un fournisseur de services a un rôle passif dans son cycle de vie et que l'objet étant demandeur de services a un rôle actif dans son cycle de vie en agissant face aux changements (attendus ou inattendus) dans son environnement. Dans le rôle passif, l'objet répond simplement aux sollicitations faites par les acteurs de la chaîne logistique (des utilisateurs, des processus, des autres produits). Nous considérons ce type de comportement comme un comportement classique (demande/réponse) dans le sens que l'objet agit comme serveur en offrant des services. Dans une approche réactive, un objet agit continuellement à de nouvelles conditions externes (inattendues) non contrôlables par l'objet. Les services demandés par des objets communicants peuvent être accessibles à partir de différentes sources telles que des bases de données distantes, d'autres objets, ou plus généralement, à partir de ressources Internet. Il est évident qu'un objet communicant évolue dans son environnement. Pour cela, il utilise et sollicite ses capacités de mémorisation, de perception, d'action et ses capacités à communiquer avec d'autres entités, afin de prendre des décisions de façon autonome pour gérer son évolution. La Figure 19 illustre le mécanisme de base : requête / réponse entre un demandeur de service et un fournisseur de services.

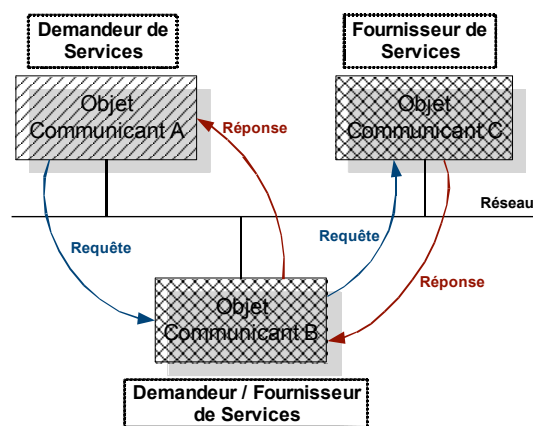


Figure 19 : L'objet comme demandeur ou fournisseur de services.

Ce mécanisme de fournisseur / demandeur de services se rapproche du comportement de *Smart Devices* [Privat, 2000]. Pour [Kärkkäinen et al., 2003] les entités qui gèrent l'information individuelle de chaque produit sont des fournisseurs de services. Dans ce cas, les produits physiques sont indirectement des fournisseurs de services accessibles depuis des applications logicielles. Dans [Smirnov et al., 2004] la notion de demandeur et fournisseur de services est utilisée pour représenter un utilisateur ou un autre type de ressource pouvant demander ou offrir des services sous des contrats particuliers afin de gérer la connaissance des processus logistiques en utilisant des services web. Ainsi, ils ont proposé un système nommé « *Knowledge Source Network* » ayant un rôle de fournisseur de services afin de présenter l'information (sur des processus logistiques) aux clients via PC, PDA, téléphone... et un rôle de demandeur de services afin de récolter dynamiquement l'information nécessaire à partir de sources externes. Dans [Riekkki et al., 2005] un objet physique équipé d'une étiquette électronique est présenté comme un fournisseur de services qui est accessible via un téléphone portable muni d'un lecteur RFID. Dans [Jarvis et al., 2006] un holon est présenté comme un demandeur et un fournisseur de services vers d'autres holons dans un système manufacturier. Dans ce dernier cas, les holons sont spécifiés par leurs services et non par leurs instances dans le système.

Comme nous l'avons précisé auparavant, le contexte est une variable importante au moment de la définition de services fournis ou demandés par un produit intelligent. Nous distinguons comme informations contextuelles pour déterminer les services d'un objet communicant : les caractéristiques des ressources informatiques, les caractéristiques des entités qui sont en relation avec l'objet communicant, et les caractéristiques de l'environnement physique. La disponibilité des ressources informatiques détermine les possibilités d'interaction entre les entités, c'est-à-dire une contrainte de fonctionnement du système. Le contexte des entités participantes pour un objet communicant est déterminé par la présence, la description, l'état et la localisation géographique de chacune des entités. Le contexte de l'environnement physique pour un objet communicant est caractérisé par les variables que le décrivent : les coordonnées géographiques, la température, l'humidité, la lumière, le bruit, etc. Cette classification est en accord avec la classification proposée par [Dey, 2000] basée sur le travail de [Schilit et al., 1994] dans lequel ils proposent trois catégories de contexte qui conditionnent une application donnée, à savoir : le contexte du système informatique, le contexte de l'utilisateur (entité), et finalement, le contexte de l'environnement physique.

Des applications plus complexes utilisent de plus en plus des capteurs pour la collecte des données décentralisées dans l'environnement et fonctionnent sous une logique de notification d'événements. Dans d'autres termes, les objets communicants informent automatiquement l'objet intéressé lorsqu'un changement d'état se produit ou si une règle préprogrammée est violée (par exemple, concernant la température ou la durée du séjour autorisée d'un objet dans un endroit donné).

En résumé, cette approche incorpore comme élément novateur la réactivité de l'objet communicant. Cette réactivité est atteinte, précisément, au moyen des interactions initiées par l'objet dans son cycle de vie dans un rôle de demandeur de services.

2.2.4 L'objet communicant sensitif

Cette approche est basée sur l'intégration physique de capteurs ou actionneurs sur ou dans l'objet physique. Ces derniers apportent à l'objet la capacité de capter des données de son environnement, de communiquer avec d'autres entités dans leur proximité, et de traiter, de façon autonome, les données localement [Siegemund, 2004]. Les objets peuvent employer des ressources de réseau local ou Internet pour effectuer leurs activités, quoique leurs utilisateurs puissent ne pas se rendre compte de ceci [Mattern, 2005]. Habituellement ils ne dépendent pas des dispositifs externes de lecture parce qu'ils sont souvent équipés d'une alimentation d'énergie autonome. Ainsi, l'objet est indépendant d'une infrastructure physique de lecteurs pour être reconnu comme un objet communicant. Puisque les capteurs sont directement intégrés dans les objets, ils peuvent à tout moment surveiller l'état d'un objet, et peuvent fournir des données plus précises sur son environnement. En conséquence, ces données peuvent être accédées sans interruption et non seulement en présence des dispositifs externes de lecture. Les objets sensitifs peuvent donc également fournir leurs services même lorsqu'aucune infrastructure n'est disponible, entre objets par exemple. Cependant, l'alimentation d'énergie des objets sensitifs est un inconvénient non négligeable. Comme exemple d'objets sensitifs nous citons la plate-forme embarquée *smart-its* proposée dans le cadre du projet *smart-its* (Cf. Figure 6). Dans ce cas, un *smart-its* est un objet communicant sensitif qui est capable de traiter l'information, d'interagir avec l'environnement physique au travers des capteurs et des actionneurs, et de communiquer avec d'autres objets via une communication sans fil [Beigl et Gellersen, 2003].

Afin de créer des objets communicants sensitifs, il est possible d'utiliser des étiquettes électroniques avec des capteurs intégrés (température, humidité, ...) pour enrichir l'information contextuelle dans laquelle les interactions sont développées [Chioiu et al., 2003] [Metras, 2005]. Les étiquettes électroniques équipées de capteurs peuvent être actives (alimentation propre) ou passives (alimentées par le champ magnétique du lecteur RFID) [Finkenzeller, 2003]. L'utilisation d'étiquettes RFID actives implique avoir des étiquettes plus grandes et plus chères que les étiquettes passives. En conséquence, les étiquettes actives sont plus difficiles à « cacher » aux utilisateurs que les étiquettes passives.

Dans cette approche nous citons les « *sentient artefacts* » [Kawsar et al., 2005], comme des objets d'usage habituel enrichis par des capteurs (pour détecter le contexte d'une interaction) et d'actionneurs afin de fournir des services avancés aux utilisateurs en plus de leur service primaire [Fujinami et al., 2005]. Le but est de déployer des fonctionnalités avec des interfaces naturelles en cachant les interfaces informatiques. Dans ce cas, l'utilisateur n'utilise aucun dispositif ou technologie consciemment. Pour eux, les objets sont utilisés de façon naturelle. Le paradigme « *Natural Interaction* »³² guide ce type d'applications.

³² www.naturalinteraction.org - *Natural Interaction Web Site* - Ing. Alessandro Valli, PhD - Italy

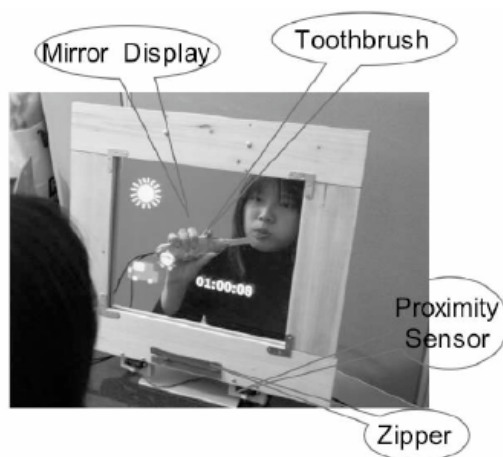


Figure 20 : Un « *sensitive artifact* » : le miroir sensible.

Comme exemple dans la Figure 20 nous montrons un « miroir sensible » développé au Japon à l'Université de Tokyo [Kawsar et *al.*, 2005]. Ce miroir est composé d'un écran d'ordinateur et de capteurs de proximité pour inférer la présence de l'utilisateur. La brosse à dents est utilisée pour activer / désactiver les services. Les services offerts à l'utilisateur, au travers du miroir, sont : l'agenda d'activités, la météo, et des informations sur le trafic. L'information montrée à l'utilisateur est obtenue au travers de services web.

2.3 La technologie d'identification automatique pour les objets communicants

Parmi les nombreuses technologies d'identification automatique d'objets (codes à barre, RFID, reconnaissance visuelle, biométrie, ...) la technologie d'identification automatique RFID est la plus utilisée dans les travaux de recherche associés au concept d'objet communicant dans les domaines informatique, télécommunication et manufacturier. Cette technologie est également fortement ancrée dans le monde industriel. Ainsi, la technologie RFID peut être utilisée comme outil pour identifier automatiquement les objets physiques, et à partir de là, il est possible de développer des nouvelles fonctionnalités pour les objets en suivant les quatre approches proposées : objet porteur de données, objets pointeur vers un système d'information (en incorporant les objets augmentés), objets fournisseur et demandeur de services et objets sensibles.

Dans nos travaux, nous avons choisi d'adopter la technologie RFID afin de permettre une intégration efficace du concept de produit communicant. La maturité de la technologie RFID [Metro Group, 2004] à ce jour permet d'atteindre de hauts niveaux de performances (taux et vitesses de lecture et écriture) dans le processus d'identification automatique à condition de bien exploiter ses avantages et de prendre en compte d'une façon adéquate ses limitations. Dans la troisième partie de ce chapitre nous expliquons la technologie RFID afin de l'utiliser correctement dans des applications industrielles et logistiques.

2.3.1 La technologie RFID

D'une façon générale, nous pouvons dire que la technologie RFID est une technologie d'identification automatique des objets par radiofréquence permettant la lecture et l'écriture des informations sur une étiquette électronique ou, plus simplement, « tag RFID » [Paret, 2001]. La RFID est apparue durant la seconde guerre mondiale dans le but de distinguer les avions amis des avions ennemis (système IFF :

identification of friend or foe). Elle est employée depuis longtemps dans le secteur agricole pour reconnaître les animaux, et plus récemment, elle a commencé à être utilisée dans l'industrie, puis la distribution et les services. La technologie RFID permet d'automatiser le processus d'identification des objets / produits afin de mieux gérer les opérations dans la chaîne logistique en minimisant les incertitudes et les erreurs humaines [Morgenroth et *al.*, 2004] [Sahin, 2004].

Un système RFID se compose de quatre éléments : un interrogateur (ou simplement antenne / lecteur RFID), un transpondeur (TRANSmetteur / réPONDÉUR) (ou étiquette électronique), un contrôleur et une application logicielle tournant sur une machine hôte [Omron, 2001]. Un lecteur RFID peut vider, renvoyer ou écrire des données sur un transpondeur. Le lecteur RFID, soit fixe soit portable, transmet un signal selon une fréquence déterminée vers une ou plusieurs étiquettes électroniques situées dans son champ d'action. Un contrôleur est un équipement physique qui permet de configurer l'utilisation du lecteur RFID afin de gérer effectivement la communication entre une étiquette et un ordinateur (logiciel) ou avec une autre machine hôte (par exemple un Automate Programmable Industriel pour exécuter des commandes d'opération). L'application logicielle sur un ordinateur agit comme une interface assurant le dialogue du contrôleur avec une étiquette électronique. Il faut remarquer qu'étant dû à la nature de l'interaction sans contact, l'air assure le médium de communication. Ainsi, l'onde électromagnétique RF effectue le transport des données. L'air réalise également le couplage magnétique entre les antennes des étiquettes et celles des lecteurs RFID. La Figure 21 illustre un système RFID dans lequel une machine hôte peut être un ordinateur fixe, un ordinateur portable ou une application industrielle. Dans ce cas, une application sur une machine hôte, utilisant un contrôleur RFID et une antenne RFID, peut exécuter des actions sur les étiquettes (lecture et écriture des données).

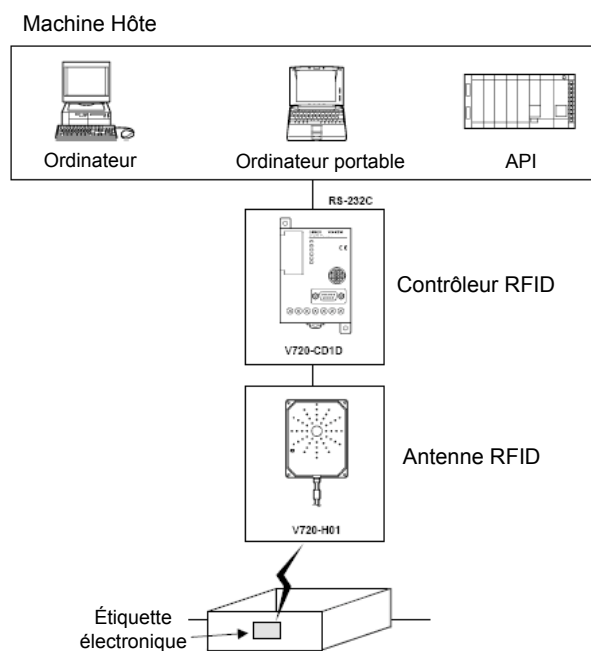


Figure 21 : Structure générale d'un système RFID.

2.3.2 L'étiquette électronique ou tag

Une étiquette électronique, également appelée « tag », est composée d'une puce en silicium reliée à une antenne (de cuivre, d'aluminium, d'encre sérigraphiée....) permettant une bonne conductivité de

l'énergie afin d'alimenter le circuit intégré (chip) [Penttilä et al., 2005]. Ceci est représenté dans la Figure 22 où l'on peut apprécier des étiquettes RFID de différentes tailles et formes : circulaire, carrée, rectangulaire et en bâton³³.

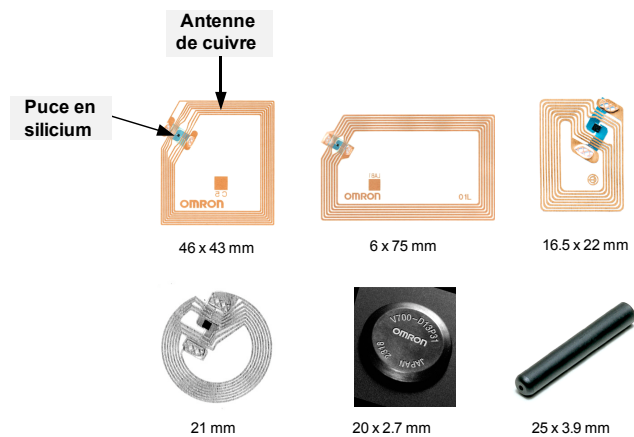


Figure 22 : Formes d'étiquettes électroniques, exemple produits Omron.

Plus précisément, une étiquette est un micro système composé de deux parties : une partie logique et une partie analogique. La partie logique incorpore la zone de mémoire, la logique de contrôle et commande de la mémoire et le traitement du signal (cryptage, décryptage éventuels ...). La partie analogique incorpore la réception, la démodulation et modulation du signal, l'alimentation et l'antenne [Paret, 2001].

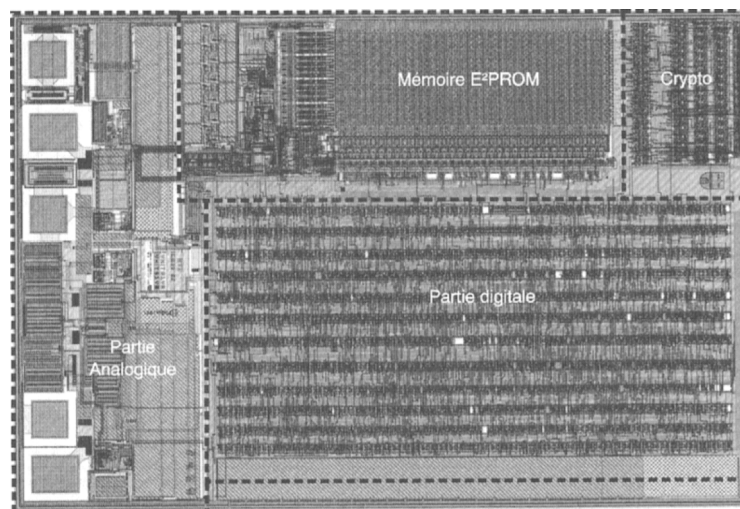


Figure 23 : Vue du silicium d'une étiquette électronique (photo Philips Semiconductors).

La Figure 23 présente une vue du silicium d'une étiquette électronique dont les fonctions sont classiques (interface analogique, logique de contrôle et de commande, mémoire E2PROM, etc.)

³³ <http://omron-industrial.com/en/home/> - Omron Industrial Automation

2.3.2.1 Etiquettes passives et étiquettes actives

En fonction de la source d'énergie qui alimente une étiquette, on peut distinguer deux types d'étiquettes (tags) : les étiquettes passives et les étiquettes actives. Un tag passif ne dispose pas de source d'énergie autonome. C'est le champ Electromagnétique émis par le lecteur, via l'antenne, qui télé-alimente une étiquette par induction magnétique dans l'antenne du tag et ainsi permet d'énergiser la structure microprocesseur du tag. La communication de données entre lecteur et tag est effectuée par modulation du champ magnétique [Finkenzeller, 2003]. Quand l'étiquette est stimulée par le lecteur, elle émet sa réponse par variation de son impédance d'entrée, modulant ainsi le couplage magnétique et la tension induite dans le lecteur en retour. Cette technique est nommée « *Backscattering* » [Keskilammi et al., 2003]. La Figure 24 illustre le mécanisme d'alimentation d'une étiquette passive.

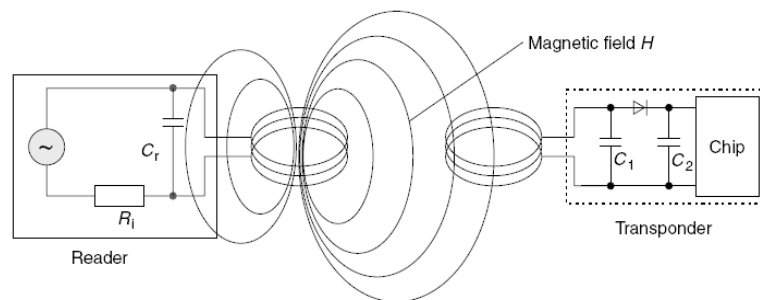


Figure 24 : Communication Lecteur – Etiquette par couplage magnétique.

Les étiquettes dites actives sont alimentées par une source d'énergie locale (pile, batterie ...). Elles permettent d'allonger la distance de lecture / écriture (de centimètres à plusieurs mètres), et parfois d'associer à l'étiquette des capteurs nécessitant une certaine quantité d'énergie pour fonctionner [Keskilammi et al., 2003]. De plus, les étiquettes actives sont de plus grande taille et plus coûteuses que les étiquettes passives. Dans les deux types d'étiquettes, aucune ligne de vue directe n'est exigée par rapport à d'autres technologies d'identification automatique telles que la technologie de code à barres ou la technologie infrarouge [Floerkemeier et al., 2003]. Selon [Vacherand, 2005] les éléments clés qui vont guider les travaux de recherche pendant les années à venir sont : l'intégration de micro capteurs dans le chip de l'étiquette, l'accroissement des capacités de traitement de l'information des chips, l'incorporation des sources d'alimentation complémentaires (micro batteries, ...) dans le chip et l'usage de micro antennes pour interagir avec les étiquettes.

2.3.2.2 Schéma fonctionnel d'une étiquette électronique

La Figure 25 illustre les fonctions de base présentes dans un transpondeur. L'interface HF permet de démoduler le signal analogique du lecteur RFID pour créer le flux de données numériques. Ce flux est traité dans le module d'adressage et sécurité. La fonction de cryptographie est optionnelle et sera utilisée pour l'authentification, la sécurité de données et l'administration de clés d'accès. L'adressage permet d'accéder à la mémoire du transpondeur. Les mémoires EEPROM (*Electric Erasable and Programmable Read Only Memory*) et FRAM (*Ferroelectric Random Access Memory*) sont utilisés pour stocker des données pour un long temps (périodes de rétention de 10 années) sans avoir besoin d'une source d'alimentation propre. La mémoire FRAM est plus performante que la mémoire EEPROM, de par une taille plus réduite, et supporte plus de cycles d'écriture, requiert moins d'énergie et permet des temps d'écriture plus intéressants. Cependant, la mémoire EEPROM est la plus utilisée

actuellement du fait du problème d'intégration de mémoire FRAM avec des microprocesseurs CMOS (*Complementary Metal Oxide Semi-conductor*) dans une unique puce [Finkenzeller, 2003].

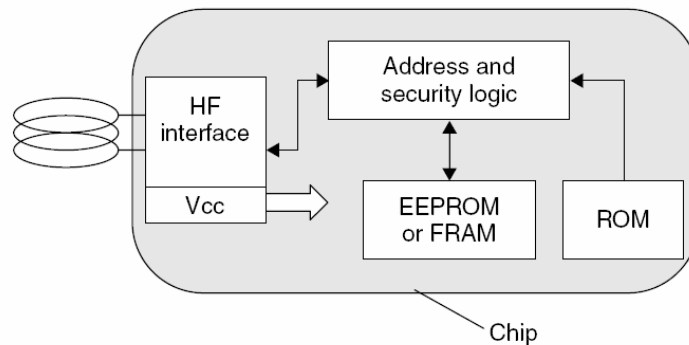


Figure 25 : Schéma fonctionnel d'un transpondeur passif avec mémoire intégré.

Pour mesurer des variables physiques, les étiquettes électroniques peuvent intégrer une grande variété de capteurs. Etant donné la consommation d'énergie d'un capteur, seuls certains capteurs sont convenables pour les étiquettes passives (sans source d'alimentation propre). Par exemple dans une étiquette passive, il est possible d'intégrer des capteurs de température, d'humidité, de pression et de lumière. Dans une étiquette active, il est possible d'intégrer les capteurs mentionnés auparavant et, en plus, des capteurs plus complexes tels que des capteurs d'accélération, de flux, de PH, de détection de gaz et de conductivité [Finkenzeller, 2003].

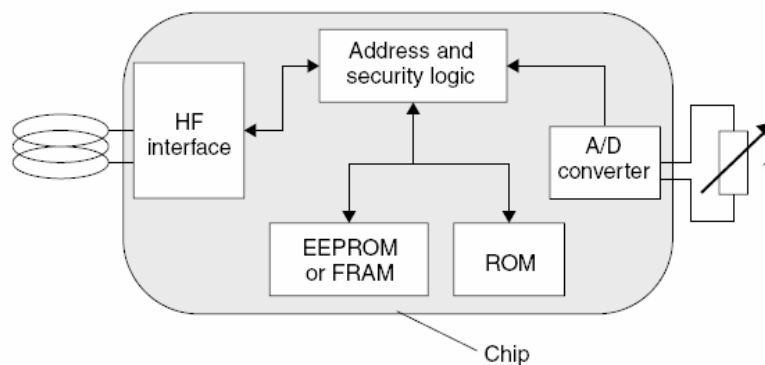


Figure 26 : Transpondeur avec un capteur de température intégré.

La Figure 26 représente le schéma fonctionnel d'une étiquette passive avec un capteur de température intégré. Le convertisseur analogique numérique (*A/D converter*) transforme le signal externe (variation de la résistance) en signal numérique pour le renvoyer au module d'adressage et sécurité de la puce. Le même principe peut être implémenté avec d'autres capteurs.

2.3.2.3 Modes de fonctionnement d'une étiquette : mode de lecture / écriture

On peut classer les différentes familles de transpondeurs selon leur mode de fonctionnement et de communication : lecture seule, lecture et écriture multiples, lecture et écriture protégées, lecture et écriture sécurisées, lecture et écriture cryptées [Paret, 2001].

- **Lecture seule** : Ce mode consiste uniquement à lire le contenu du transpondeur en supposant que celui-ci contienne quelque chose de préalablement inscrit. On distingue deux possibilités : le transpondeur a été préalablement écrit par le fabricant du composant (numéro unique, contenu personnalisé, ...) ou le transpondeur, livré vierge par le fabricant, a été écrit de façon unique (WORM : *Write Once, Read Multiple*) par l'utilisateur, selon ses besoins. Une fois cela effectué, le transpondeur devient alors accessible en « lecture seule ».

- **Lecture et écriture multiples** : Ce mode de fonctionnement permet la réutilisation, la réinscriptibilité du transpondeur. On distingue deux possibilités.

Le premier mode est le **MTP** (*Multi times programmable*) permettant d'effectuer certaines modifications particulières du contenu des données sur le transpondeur un nombre limité de fois, de type de mise à jour, version, etc.. Le but principal du transpondeur, par la suite, n'est que d'être lu.

Le deuxième mode est le **R/W** (*Read / Write*). Ici, la réinscriptibilité multiple du transpondeur est permise sans limite du nombre de ré-écritures. Les cycles d'écriture pour la mémoire EEPROM ont un ordre de grandeur de 10 000 à 100 000 [Finkenzeller, 2003]. Ce type d'étiquette peut être employé, par exemple, sur des chaînes de production pour assurer le suivi des interventions ayant lieu sur des produits tout au long de leur élaboration.

- **Lecture et écriture protégées** : Le mode d'utilisation « lectures et écritures protégées » concerne la lecture de tout ou partie du contenu du transpondeur protégé sous condition (mot de passe...), ainsi que la modification du contenu sous les mêmes conditions. Le lecteur RFID, pour avoir un droit d'accès, doit justifier « l'autorisation » au transpondeur sous quelque forme que ce soit (mot de passe, procédure hardware, timing particulier...), etc. L'écriture protégée est utilisée pour sécuriser l'écriture du contenu de la mémoire : soit en créant des zones de la mémoire fonctionnant en mode OTP (*One Time Programmable*) ; soit après de multiples modifications volontaires du contenu, en décidant de verrouiller définitivement une zone mémoire contre toute prochaine écriture; soit en permettant l'accès à la réécriture de tout ou partie de la mémoire seulement sous condition(s) spécifique(s), ici aussi sous « x » mots de passe plus ou moins complexes.

- **Lecture et écriture sécurisées** : Ce mode de fonctionnement permet, à l'aide de techniques classiques de sécurisation (authentification de partenaires, code évolutif ou tournant par exemple), d'atteindre un niveau plus élevé de sécurisation de l'établissement de la communication entre le lecteur RFID et le transpondeur.

- **Lecture et écriture cryptées** : De même, le contenu de la communication entre le transpondeur et le lecteur peut être crypté de façon à déjouer des écoutes pirates des communications échangées. Ceci est généralement effectué à l'aide d'algorithmes cryptographiques de type soit propriétaires, soit à clés privées (par exemple DES : *Data Encryption Standard* dans sa version actuelle 3DES), soit à clés publiques (par exemple RSA : *Rivest, Shamir, et Adleman*).

Dans le but de respecter la confidentialité des transactions entre un lecteur RFID et une étiquette électronique [Floerkemeier et Langheinrich, 2004] proposent d'incorporer aux actuels standards les opérations suivantes : l'identification obligatoire de l'interrogateur d'une étiquette (ID lecteur), la spécification de la nature de ces interactions (en fonction d'une typologie), et la limitation des interactions concernant des données personnelles en fonction d'un profil standardisé.

2.3.3 Distance de communication

La notion de distance de communication permet de classer les applications sans contact en fonction de la distance entre l'interrogateur et le transpondeur [Paret, 2001]. Selon l'organisme international de normalisation (ISO), les applications dites de « courtes distances » (Norme ISO 10536) sont celles fonctionnant en général sur des distances de l'ordre de quelques millimètres ou dizaines de millimètres. La notion d'accès sans contact de « proximité » ou « *Proximity* » (Norme ISO 14443) représente des distances de l'ordre de la dizaine de centimètres. Pour le sans contact de « voisinage » ou « *Vicinity* » (Norme ISO 15693), les distances sont de l'ordre de 50 à 70 centimètres. Des applications fonctionnant sur des distances de l'ordre du mètre (0,8 à 1,5 m) sont dites « de longue distance ». Les applications dites « de très longue distance » fonctionnent sur de distances supérieures à quelques mètres...jusqu'à quelques dizaines, voire centaines de mètres.

2.3.4 Fréquences de communication

La transmission de données se fait selon diverses fréquences classées en quatre groupes dans le spectre ISM (*Industriel Scientifique et Médical*) géré par l'ARCEP en France (Autorité de Régulation des Communications Electroniques et des Postes) :

- les « basses fréquences » dites LF inférieures à 135 KHz (deux fréquences sont principalement utilisées, le 125 KHz et le 134 KHz) ;
- les « hautes fréquences » dites HF avec une fréquence utilisée majeure de 13.56 MHz ;
- les « ultra hautes fréquences » dites UHF avec deux fréquences utilisées, le 433 MHz et la bande allant de 864 à 868 MHz ouverte en France en 2006. Les USA et l'Asie travaillent dans la bande 860-915 MHz.
- les « micro-ondes » dites VHF avec deux fréquences réservées, 2.45 GHz et 5.8 GHz.

Il faut préciser que chaque fréquence présente des caractéristiques différentes, tant du point de vue des paramètres de communication (distance, vitesse d'échange) que vis-à-vis de la robustesse aux perturbations liées à l'environnement dans lequel elle fonctionne (présence de métal, présence de liquide, champ électromagnétique, ...).

Le Tableau 2 montre une analyse comparative des performances caractéristiques des différentes fréquences [d'après Paret, 2003]. L'influence du métal, de l'eau ou du corps humain dépend de la fréquence de la porteuse. Par exemple, pour une porteuse de 13.56 MHz, la présence d'un objet métallique à 50 mm du transporteur entraîne une atténuation de 90% du signal. Il est clair que chaque fréquence a son propre domaine d'application préférentiel en fonction de la distance de lecture et écriture souhaitée et le débit de communication nécessaire. A contrario, certains domaines d'applications peuvent être couverts par plusieurs fréquences de travail.

Caractéristiques / Fréquences	< 135 kHz		13.56 MHz		860 / 930 MHz		2.45 GHz	
Type de fréquence	Basse		Haute		Bandes UHF		Hyper	
Technologie d'antenne (impact sur coût)	Bobine à air ou sur ferrite	B	Boucle imprimée perforée Gravée	E	Boucle imprimée perforée Gravée	B	Antenne imprimée gravée	E
Distance de lecture et écriture	> 1 m	B	Europe / France : Max : 1 m US>0.8 m	B	Europe / France : Pour le moment pas utilisable fréquences GSM. Télécom, ...	-	France < 0.1 m (en télé alimentée) (> 1 m avec dérogation)	M
					US> 1 m à 10 m	E	US> 1 m	
Vitesse de transfert des données	< 10 Kbit/s	M	> 100 kbit/s	B	> 100 kbit/s	B	> 200 kbit/s	E
Influence métal	perturbation (Espace > 50 mm = 90% Perf.)	M	perturbation (Espace > 50 mm = 90% Perf.)	M	perturbation (Espace > 50 mm = 100% Perf.)	B	perturbation (Espace > 5 à 7 mm = 100% Perf.)	B
Influence de l'eau	aucune	E	Atténuation	B	Atténuation	B	perturbation	M
Influence de corps humain	aucune	E	Atténuation	B	Atténuation	B	perturbation	M

Performances : **M** = Médiocres, **B**= Bonnes, **E**=Excellentes.

Tableau 2 : Résumé des principales performances des différentes fréquences RFID.

2.3.5 Temps de communication

Le choix d'une solution technique RFID pour une application déterminée exige de définir les conditions d'utilisation de la technologie : le temps de transaction entre le lecteur RFID et l'étiquette, la vitesse de mobilité d'une étiquette par rapport à un lecteur RFID, le nombre d'étiquettes présentes simultanément dans le champ magnétique. Dans cette partie, nous examinerons la durée d'une transaction entre une antenne et un transpondeur. Dans le point suivant, les deux éléments restants seront examinés.

[Paret, 2001] signale que le temps d'une transaction est fonction :

- du temps de réveil et du temps de reset que met le transpondeur pour être en état de travailler après être entré dans le volume d'action du champ magnétique, généralement ce temps est de l'ordre de quelques millisecondes ;
- du temps de gestion d'hypothétiques collisions (procédure d'anticollision) dues à la présence simultanée de plusieurs transpondeurs dans le champ ainsi que des temps nécessaires aux authentifications, cryptages, décryptages ;
- de la somme des temps de retournements émission/réception/émission/réception/etc. nécessaires aux lecteurs et transpondeurs pour fonctionner correctement ;
- de la quantité d'informations à échanger dues à l'application et de leur gestion (lecture, écriture en mémoire EEPROM, calculs sur les données présentes dans le transpondeur, etc.)
- du débit brut de l'échange, incluant bien évidemment la structure protocolaire de la communication (en-tête, prologue, épilogue, CRC, etc. du protocole de communication) ;
- du temps de la présence physique du transpondeur dans le champ magnétique en fonction ou non de son déplacement relatif dans le temps par rapport au lecteur ;
- etc.

Fonctions	Temps passé pour effectuer la fonction considérée
Reset	Environ 5 ms
Answer to request et séquence d'anticollision	3 ms
Sélection	2 ms
Authentification	2 ms
Lecture de 16 octets	2,5 ms
Ecriture de 16 octets + relecture de contrôle	9 ms
Total	Environ 25 ms

Tableau 3 : Temps de communication de tag MIFARE de Philips.

Le Tableau 3, à titre d'exemple, indique le cumul de ces temps élémentaires pour les produits MIFARE de Philips Semiconductors fonctionnant à 13.56 MHz avec un débit de communication de 106 Kbits/s [Paret, 2001]. Dans ce cas, le temps total estimé de communication pour lire puis écrire 16 octets, est de 25 ms.

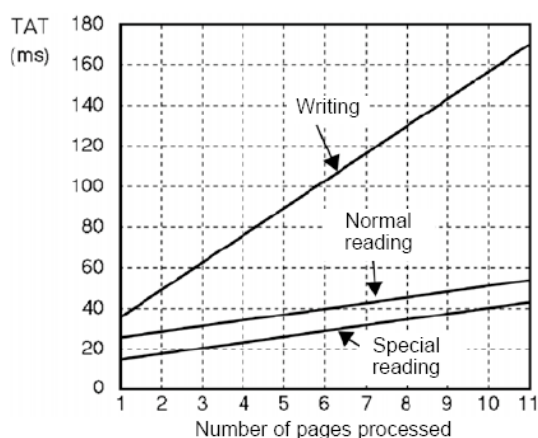


Figure 27 : Temps de transaction Lecteur – Etiquette selon les données accédées.

La Figure 27 est basée sur un rapport technique élaboré par [Omron, 2001] indiquant le temps de communication attendu pour un système RFID composé d'étiquettes HF 13.56 MHz du type Icode1 Philips. On peut observer la relation entre le temps total de communication (en ms) lecteur-étiquette (TAT : *turn around time*) et le numéro de pages de mémoire traités dans une opération d'écriture ou de lecture. Chaque page mémoire contient 4 octets. Le graphique montre que le temps d'écriture est beaucoup plus élevé que le temps de lecture pour tous les volumes de données échangés. Ceci est justifié, principalement, par le fait que lorsque l'on écrit dans la mémoire, on effectue une relecture de ce que l'on vient d'écrire (un *READ* after *WRITE*) afin de s'assurer que l'on a bien écrit ce que l'on voulait et que le transpondeur ne s'est pas échappé du champ pendant son temps d'écriture théorique [Paret, 2001].

En résumé, il existe une forte implication entre la valeur de la fréquence porteuse, le débit de communication (Kbits/s), le protocole de communication et le temps global que dure une transaction.

2.3.6 Méthodes de communication

Les méthodes de communication courants entre lecteur et tag sont : lecture et écriture d'une étiquette unique dans le champ magnétique, lecture et écriture FIFO (*first-in, first-out*), lecture et écriture multiple, lecture et écriture sélective [Omron, 2001]. (Cf. Figure 28).

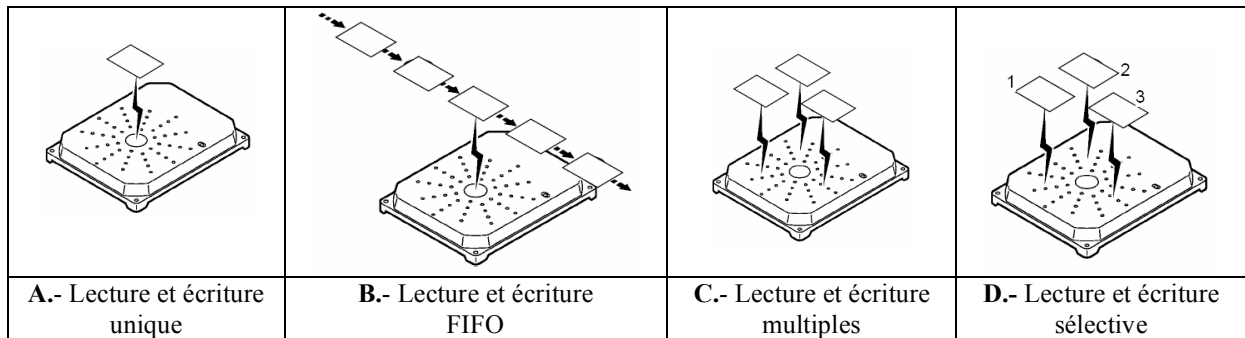


Figure 28 : Méthodes d'accès aux étiquettes électroniques.

A.- Lecture et écriture d'une étiquette unique présente dans le champ magnétique :

Dans cette méthode une seule étiquette doit être présente dans le champ d'action du lecteur.

B.- Lecture et écriture selon une règle FIFO :

Dans cette méthode, le lecteur RFID lit et écrit des données sur chaque étiquette entrant dans la zone de communication l'une après l'autre. Une fois que chaque étiquette est traitée, son accès est bloqué temporellement. La communication avec l'étiquette sera possible seulement si elle arrive nouvellement à la zone communication de l'antenne. Dans ce cas, la vitesse maximale de déplacement d'une étiquette peut être calculée facilement en utilisant la formule :

Vitesse maximale = (distance de référence / temps de communication) * facteur de sécurité³⁴

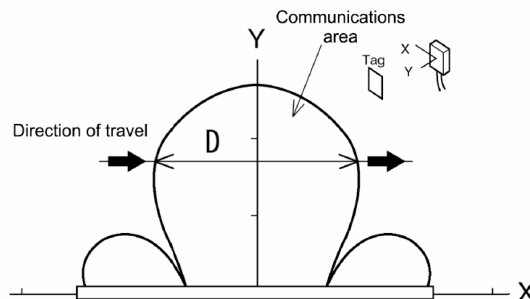


Figure 29 : Lobe de rayonnement d'une antenne RFID 13.56 Mhz.

La Figure 29 montre le lobe de rayonnement électromagnétique d'une antenne RFID HF et la distance de référence « D » utilisée pour calculer la vitesse maximale de passage du tag. Par exemple, pour un objet à une distance de référence (champ électromagnétique du lecteur) de 25 cm, un temps de lecture (de 12 octets) de 25 ms, et un facteur de sécurité de 50%, la vitesse maximale de déplacement de l'objet est de 5 m/s, soit 18 km/h. Cette vitesse paraît raisonnable pour des applications de type industriel.

C.- Lecture et écriture d'étiquettes multiples dans le champ magnétique :

Il se peut aussi que plusieurs étiquettes soient présentes simultanément (volontairement ou involontairement) dans le volume dans lequel se développe le rayonnement du lecteur. Le système doit être capable d'identifier toutes les étiquettes présentes dans le champ magnétique du lecteur.

³⁴ Omron Corporation recommande employer un facteur de sécurité égal à 0.5 (50%) [Omron, 2001]

La Figure 30 nous montre le temps de communication entre un contrôleur RFID et des étiquettes multiples afin de lire et écrire 8 octets simultanément des données sur un ensemble d'étiquettes. Il est important de mentionner que dans tous le cas (lecture / écriture) le temps de communication est directement proportionnel à la quantité de étiquettes traitées³⁵. Le temps total de communication (hôte - étiquette) doit être obtenu en additionnant aux valeurs signalées le temps de communication hôte - lecteur (requête / réponse).

Number of Tags	Timeslot setting (Number of Timeslot)	2 page (8 bytes) to read (ms)		2 page (8 bytes) to write (ms)	
		Standard mode	Fast mode	Standard mode	Fast mode
2	1	560	144	715	190
4	2	917	301	1139	378
8	3	1659	644	2016	784
16	4	3239	1391	3867	1656

Figure 30 : Temps de communication avec des étiquettes électronique radiofréquences.

D.- Lecture et écriture sélective d'étiquettes :

Dans cette méthode, le lecteur RFID doit être capable de dialoguer individuellement avec tout ou partie des étiquettes, par exemple un ou plusieurs sous ensembles d'étiquettes ayant une ou plusieurs particularités communes.

2.3.7 Modèle de communication en couche pour la RFID

Afin d'assurer une compatibilité fonctionnelle entre équipements hétérogènes, les membres de l'ISO ont proposé un jeu de normes ayant pour but de définir l'interconnexion et le comportement de chaque équipement vis-à-vis de son environnement fonctionnel. Le document de l'ISO n° 7498 de novembre de 1984 « *Information Processing Systems, OSI pour Open Systems Interconnection Basic Reference Model* », ou « modèle ISO/OSI », définit un modèle décrivant le découpage de tâches et une formalisation des flux d'échanges entre celles-ci, en particulier pour les systèmes communicants. Le modèle ISO/OSI très détaillé décrit dans la norme est composé de 7 couches superposées, numérotées de 1 à 7 [ISO OSI, 1994]. Très souvent, pour assurer une communication, il n'est pas nécessaire d'écrire et/ou remplir toutes les conditions décrites dans toutes les couches [Paret, 2001]. Dans ce cas, on parle d'un modèle « écrasé ».

Un modèle de communication adapté aux communications RFID a besoin d'un protocole de communication (ou d'un langage standardisé) pour faire dialoguer les lecteurs RFID et les étiquettes électroniques. Comme tout langage il comprend un vocabulaire et une syntaxe couvrant les commandes et les données contenues dans les étiquettes. En considérant comme référence le modèle ISO/OSI, on peut distinguer trois couches principales : **la couche application**, c'est-à-dire, les données et les messages échangés pour satisfaire le processus métier de l'application terminale, **la couche liaison de données**, c'est-à-dire, comment les étiquettes et le lecteur dialoguent et se comprennent, et **la couche physique**, c'est-à-dire, la gestion de l'interface aérienne [Knospe et Pohl, 2004]. La Figure 31 présente le modèle de communication entre un lecteur RFID et une étiquette électronique en incluant les trois couches indiquées.

³⁵ Dans ces quatre cas, la relation entre les deux variables (temps de communication et quantité d'étiquettes) s'ajuste à une ligne droite avec un coefficient de détermination R² égal à 99,9X %.

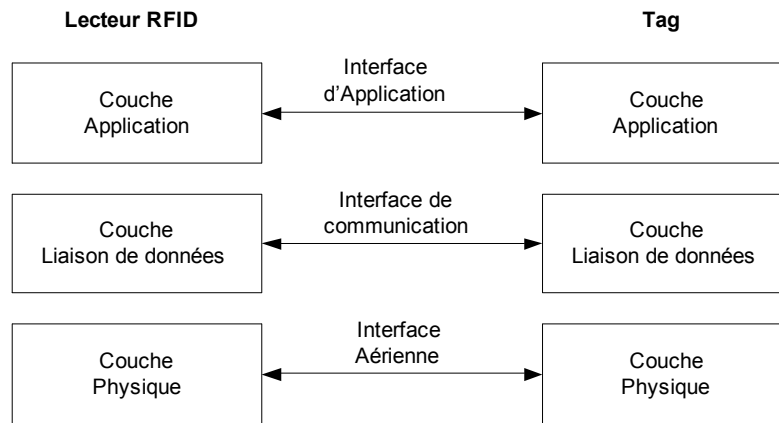


Figure 31 : Modèle de communication RFID basé sur le modèle OSI.

La **couche application** comprend : le processus métier qui est du seul ressort de l'entreprise ; le traitement informatique correspondant qui est lié aux logiciels utilisés, au système d'exploitation et au réseau de communication ; les données échangées avec l'étiquette qui peuvent être individuelles ou codifiées dans des applications en boucle fermée ou ouverte (transport ou commerce) ; la syntaxe utilisée pour structurer les données lors de l'échange. La normalisation est obligatoire dans un contexte multi-acteurs. La **couche liaison de données** a pour but de gérer la reconnaissance et l'identification d'une ou plusieurs étiquettes entrant dans le champ du lecteur et de gérer l'algorithme d'anticollision permettant de dialoguer avec plusieurs étiquettes dans le même champ. Cette couche comprend : la présentation des données dans l'étiquette ; la taille de la mémoire de l'étiquette ; la lecture de tout ou partie du contenu ; l'écriture sur l'étiquette ; la sécurité de l'échange ; l'intégralité des données échangées ; l'autorisation de lecture et écriture ; la protection de données contenues ; la protection lors du transfert des données et la fin du dialogue. Enfin, la **couche physique** considère la définition des caractéristiques des étiquettes et des lecteurs RFID pour permettre le dialogue. Cette couche a la responsabilité de spécifier : la fréquence ; la modulation (fréquence ou amplitude) ; la vitesse de transmission (kbps). De toutes ces caractéristiques dépendent : la distance de lecture ; le volume du champ de lecture ; le volume d'informations échangées ; la pollution électromagnétique des équipements environnants, en particulier d'autres étiquettes et lecteurs. La fréquence est l'élément clé dans cette couche.

2.3.8 Standards pour la RFID

La définition d'un standard dans le domaine de la RFID a pour but d'assurer l'interopérabilité entre les systèmes et les technologies. Selon [Kintzig et al., 2002] l'interopérabilité est la capacité des systèmes initialement indépendants et différents, à pouvoir fonctionner en interaction. Afin de comprendre l'aspect normatif nous pouvons distinguer les normes dites « technologiques » des normes dites « applicatives ». Par normes « technologiques » nous entendons toutes les normes qui concernent la communication entre lecteurs RFID et étiquettes ainsi que la gestion de données contenues dans les étiquettes. D'une part, les standards technologiques établissent les bases techniques d'un système d'identification automatique RFID en définissant les fréquences, les vitesses de transmission, les délais, les codifications, les protocoles et les systèmes anticollision [Dessenne, 2005]. D'autre part, les normes « applicatives » sont des normes fixées par des catégories d'utilisateurs des standards technologiques. Dans ce sens, la traçabilité de objets est une application qui constitue un domaine en pleine effervescence.

Précisons qu'en ce qui concerne les normes technologiques, l'instance normative n'est pas l'ISO mais un *Joint Technical Committee* (JTC) constitué à partir de l'ISO et de l'IEC. Il s'agit du ISO/IEC/JTC1. Le *Sub-Committee* SC31 gère la traçabilité des objets. Le SC31 a réparti les tâches entre quatre *Working groups* : Le WG2 (*Work Group on Data Structure*), le WG3 (*Work Group on Conformance*), le WG4 (*Work Group on RFID Item Management*), le WG5 (il s'occupe de la géolocalisation en temps réel, ou *Real Time Locating System - RTLS*). En France le CN31 de l'AFNOR (Association Française de Normalisation) gère la traçabilité d'objets au niveau national.

Les normes dites « ISO 18000 », publiées récemment, sont destinées à la gestion des articles. Elles sont basées sur un protocole de communication commun entre lecteur et étiquette permettant d'organiser de façon unique la structure des données contenues dans les étiquettes. Les normes sont déclinées par fréquence. On distingue donc les normes ISO/IEC 18000 [ISO 18000-1, 2004] avec cinq parties afin de détailler les paramètres de communication dans les fréquences 135 KHz [ISO 18000-2, 2004], 13.56 MHz [ISO 18000-3, 2004], 2.45 GHz [ISO 18000-4, 2004], UHF de 860 à 960 MHz [ISO 18000-6, 2004], 433 MHz [ISO 18000-7, 2004]. Il faut indiquer que les normes techniques ont été construites sur des propositions de fournisseurs de technologies. Complémentairement, l'ISO afin de vérifier la conformité des produits mis sur le marché a généré les rapports techniques 18047 [ISO 18047-3, 2004] [ISO 18047-4, 2004] [ISO 18047-7, 2005] qui se déclinent comme les normes de base, c'est-à-dire, par fréquence. Pour assurer la cohérence des commandes de lecture et la gestion de données dans le transpondeurs, trois normes ont été publiées : les normes ISO/IEC [ISO 15961, 2004], [ISO 15962, 2004] et [ISO 15963, 2004]. La norme ISO/IEC 15961 se charge du transfert des données de et vers les unités de lecture. La norme ISO/IEC 15962 décrit l'organisation et le traitement de données dans le transpondeur. La norme ISO/IEC 15963 assure que toute étiquette électronique RFID aura un numéro unique. Cette unicité permet une identification et une traçabilité de chacune des étiquettes. De plus, les normes ISO/IEC [ISO 19762-1, 2005] [ISO 19762-2, 2005] [ISO 19762-3, 2005] incluent le vocabulaire utilisé dans toutes les normes ISO traitant de RFID. Le rapport techniques ISO/IEC [ISO 18046, 2005] permet aux intégrateurs de solutions RFID de trouver les systèmes adaptés aux besoins de leurs clients sur la base de performances vérifiées et qui peuvent constituer un référentiel même si les test ne sont pas pratiqués « in situ ».

Les standards applicatifs ont pour but de conseiller certaines solutions technologiques en fonction des différentes applications ou de secteurs d'application [Walk, 2004]. Nous pouvons citer la norme allemande [VDI 4472, 2005] qui constitue un standard pour l'utilisation des transpondeurs dans l'industrie textile. A signaler également que le comité TC 104 et TC 122 travaillent à la création d'un standard dans le domaine logistique en utilisant la fréquence UHF : normes ISO 17363 à 17367 (*Draft International Standards*). Les normes [ISO 11784, 1996], [ISO 11785, 1996] et [ISO 14223-1, 2003] sont utilisées pour l'identification des animaux (fréquence à 135 kHz). Le standard EPC peut être classé comme norme applicative. Le standard EPC a donné origine à une nouvelle norme : ISO 18000-6 version C.

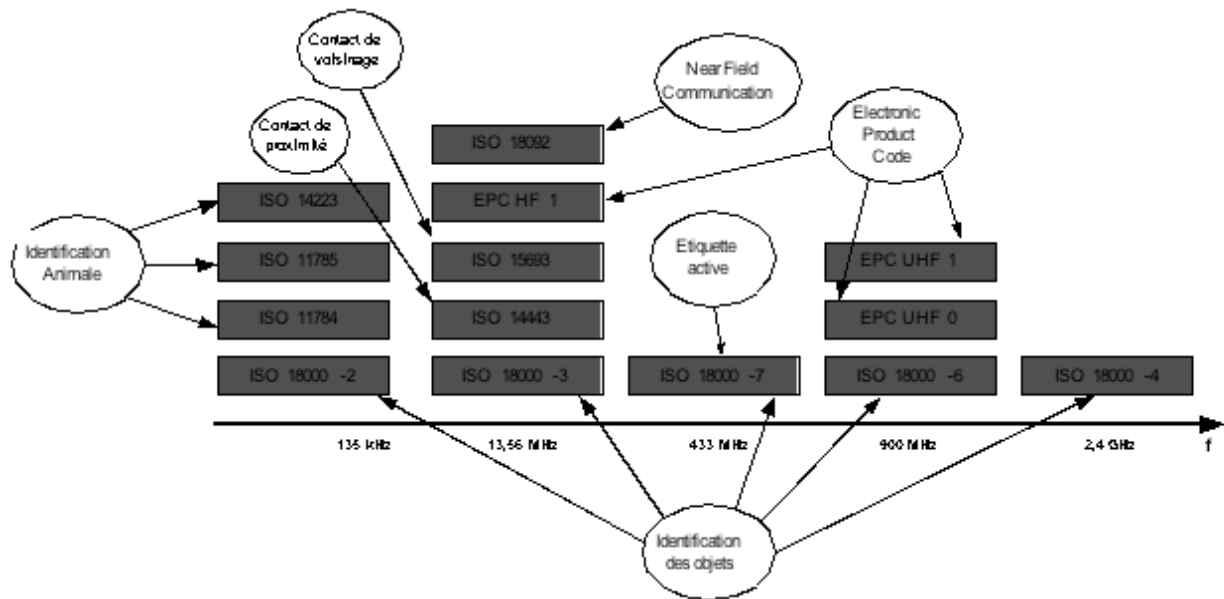


Figure 32 : Normes RFID et bandes de fréquences.

La Figure 32 [Knospe et Pohl, 2004], illustre la répartition des normes technologiques et des normes applicatives pour les différentes bandes de fréquences. Par exemple, les normes applicatives pour l'identification animale présentent la même fréquence de communication (135 KHz) que les normes technologiques [ISO 18000-2, 2004].

2.3.9 Contraintes dans l'usage des étiquettes radio fréquence

Il est possible que la présence d'une étiquette dans le champ magnétique du lecteur RFID ne soit pas reconnue selon son orientation par rapport à la surface du lecteur [Omron, 2001] [Clarke et al., 2006].

La Figure 33 montre l'influence de l'angle de l'étiquette par rapport à l'axe perpendiculaire à l'antenne RFID sur la distance de communication. Les valeurs indiquées sont valables pour un système RFID Omron opérant avec des étiquettes Icode1 13.56 Mhz. Cependant, les valeurs limites sont valides dans tous les cas. C'est-à-dire, que quand la surface de l'étiquette et la surface de l'antenne sont parallèles, on obtient une distance de communication maximale. Au contraire, quand la surface de l'étiquette est perpendiculaire à la surface de l'antenne, il n'y a pas de communication possible.

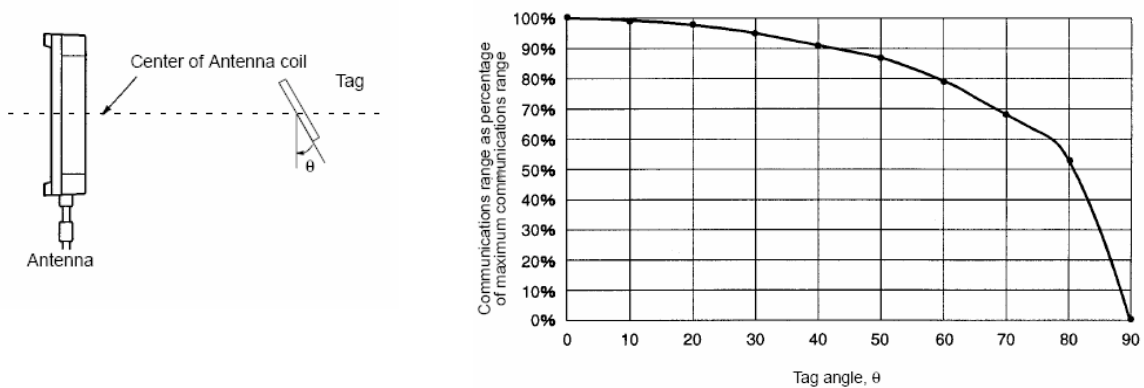


Figure 33 : Influence du positionnement Etiquette / Antenne.

La perturbation par l'environnement physique, comme par exemple, la présence de métaux dans l'environnement immédiat d'une étiquette électronique, peut perturber la communication entre le lecteur RFID et l'étiquette [Omron, 2001]. Les facteurs influant sur la communication RFID sont la taille et la forme de l'objet métallique et la distance entre cet objet métallique et l'étiquette RFID [Sydänheimo et al., 2005]. La Figure 34 montre le taux de diminution de la distance de communication quand le métal est à une certaine distance du dos d'une étiquette 13.56 Mhz. L'axe vertical du diagramme indique la distance entre une étiquette et la plaque métallique tandis que l'axe horizontal indique le taux de diminution de la distance de communication. Le diagramme montre que la communication est nulle quand l'étiquette est montée sur le métal. Cependant, il y a des systèmes RFID adaptés pour opérer dans cette situation. Par exemple [Sydänheimo et al., 2005] ont testé deux prototypes d'étiquettes passives attachées à des objets métalliques de différentes tailles et formes en obtenant des distances de communication de l'ordre de 1 mètre.

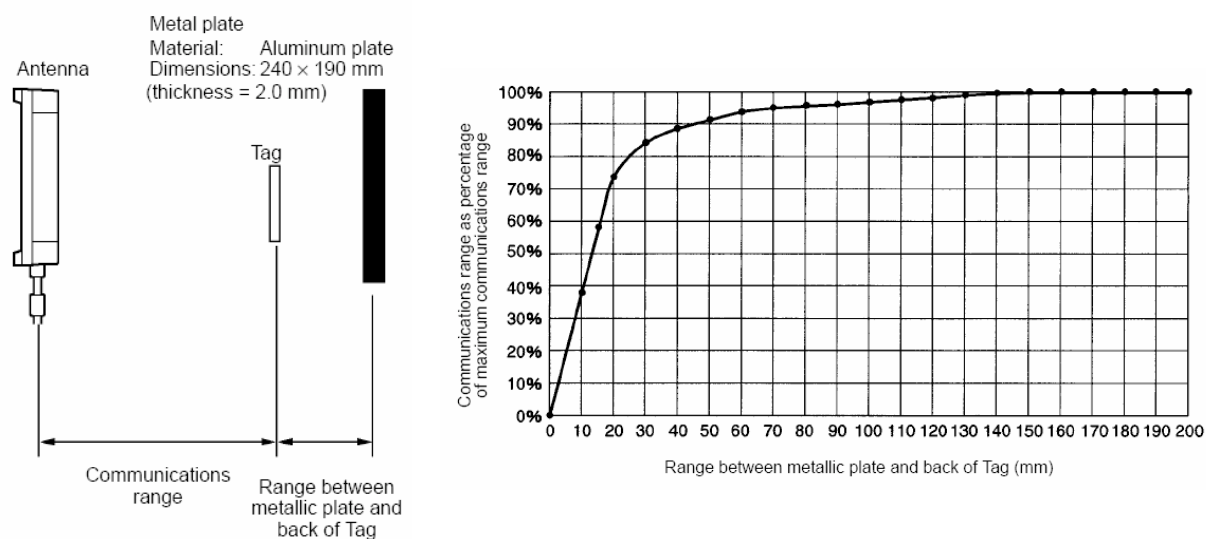


Figure 34 : Influence de l'environnement métallique sur l'accès à une étiquette.

L'influence de l'eau sur un système RFID (915 Mhz) a été testée par [Clarke et al, 2006]. Dans ce cas, ils ont démontré que la présence de l'eau dans des bouteilles dotées d'étiquette électronique affecte significativement la performance de la lecture en fonction de l'orientation de l'étiquette par rapport à la surface du lecteur. La présence d'équipements informatiques ou d'équipements électriques à proximité d'un système RFID peut entraîner des interférences dans la communication due aux ondes électromagnétiques parasites.

Nous constatons que la nécessité d'une bonne compréhension théorique des contraintes de la technologie est fondamentale. Des méthodes standardisées pour tester les différentes configurations sont nécessaires. Dans ce sens, l'analyse expérimentale est un outil pour tester et comparer les performances d'un système RFID.

2.4 La technologie RFID et les objets communicants

Nous remarquons que les étiquettes électroniques, notamment les passives, sont idéalement discrètes et « invisibles » aux utilisateurs à un degré tel qu'elles ne dérangent pas la façon dont les gens les emploient normalement. L'usage des étiquettes électroniques doit simplement ajouter une fonctionnalité additionnelle aux objets sans déranger les habitudes des personnes qu'interagissent avec

elles. Ainsi, en attachant de petites étiquettes électroniques aux objets physiques, ces objets peuvent être automatiquement identifiés et localisés une fois introduits dans la proximité d'un système de détection d'étiquettes. Concrètement, les étiquettes électroniques permettent de faire le lien entre le monde physique (objet physique) et le monde virtuel (représentation informationnelle de l'objet) en permettant, en même temps, l'exécution d'un éventail d'applications ubiquistes [Römer et *al.*, 2004]. Opérationnellement, dès que les systèmes de RFID détectent la présence d'une étiquette électronique dans un certain espace physique, le lien peut être établi en configurant les services / applications disponibles dans l'infrastructure physique. De plus, afin d'intégrer l'information contextuelle d'un objet communicant il est même possible d'utiliser des étiquettes électroniques (actives ou passives) avec des capteurs intégrés (température, pression, humidité, ...) [Metras, 2005].

Selon [Kintzig et *al.*, 2002], dans le monde industriel les étiquettes permettraient d'assurer le suivi complet des objets, un par un, depuis leur fabrication jusqu'à leur recyclage en passant par leur distribution, permettant une gestion totalement intégrée avec les outils informatique d'ERP (*Enterprise Resource Planning*), suivi de la production et la gestion de stocks. L'idée de renforcer le couplage entre le monde informationnel et le monde physique, qui est l'essence même de la révolution des objets communicants, trouve ici son aboutissement ultime et parfaitement concret.

2.5 Conclusions

L'état de l'art nous a permis de connaître les principales approches adoptées par la communauté scientifique afin de caractériser un objet communicant. Nous pouvons affirmer que le concept d'objet communicant, de *Smart Object* et de produit intelligent sont équivalents quant à la vision d'apporter de nouvelles capacités aux objets physiques. Chacun de ces concepts provient de domaines différents respectivement : informatique, télécommunications et automatique. Cependant dans les trois domaines la technologie RFID est employée comme une technologie adéquate pour identifier automatiquement un objet. Dans ce sens, il existe un consensus sur le fait que l'identification automatique est la première exigence avant de proposer des capacités additionnelles aux objets. Dépendant de la nature de l'objet et de son contexte, de nouveaux services peuvent être développés selon une approche ubiquiste. Un objet est caractérisé comme prestataire / demandeur de services. Les interactions requièrent une architecture informatique pour permettre aux objets communicants de s'exprimer et de montrer leurs propriétés et fonctionnalités aux acteurs intéressés.

La technologie RFID permet d'identifier automatiquement un objet communicant de façon discrète. L'identification de l'objet au moyen d'un identifiant unique garantit l'interaction individuelle et personnalisée avec chaque objet. Une étiquette électronique peut stocker, en plus d'un identifiant, des données conséquentes sur l'objet. L'identifiant unique de l'objet permet de lier sa partie physique avec sa partie informationnelle afin d'exploiter de nouvelles capacités. Ainsi, l'objet est en communication permanente avec des ressources informatiques distribuées dans un réseau de communication. La technologie RFID permet d'enrichir un objet, voire de le rendre plus actif et intelligent dans son cycle de vie. Des mécanismes d'action et de décision de l'objet doivent être définis pour des contextes différents.

Les applications industrielles de la technologie RFID sont associées principalement à l'identification et traçabilité d'objets [Paret, 2001] [Paret, 2003] [Sahin, 2004] [Wang et *al.*, 2006]. Les applications plus connues sont l'identification des objets, dans un sens général (animaux, paquets, bagages, vêtements, ...), le suivi d'objets (produits, cartons, palettes, bouteilles, tableaux, containers, courrier, paquets, ...) dans la chaîne de distribution et la gestion de stocks (inventaire et location d'objets) Comme application innovatrice, nous développons dans notre thèse la génération de services ambiants

pour des produits communicants en utilisant la technologie RFID [Cea et Bajic, 2004] [Bajic et Cea, 2005].

Nous constatons que les services d'un objet communicant sont de nature ambiante. C'est-à-dire, ces services entourent les objets communicants dans leurs dimensions géographiques et temporelles. L'exécution de ces services concrétise le concept d'interaction des objets. Pour mettre en place cette approche de services ambiants, il est nécessaire de définir une architecture de services afin d'organiser les éléments composant le système informatique en intégrant les objets communicants et les acteurs potentiels de la chaîne logistique. Cette architecture doit permettre d'annoncer les services de chaque objet de manière automatique, et quasi instantanée, vers les acteurs intéressés et de transmettre des requêtes entre acteurs afin d'obtenir des informations sur un objet, ou d'invoquer une méthode spécifique sur l'objet, en utilisant des technologies Internet. Ces interactions automatiques entre acteurs donnent lieu à une connaissance mutuelle entre objets. Le chapitre suivant traite cette problématique en identifiant les fonctionnalités nécessaires pour faire interagir les objets communicants dans leur cycle de vie avec des entités représentant les acteurs de la chaîne logistique.

3 Architectures de Services Ambiants

Dans ce chapitre nous analysons l'application du concept de service ambiant dans le contexte des systèmes de production et de la chaîne logistique. En général, nous pouvons dire que les services ambiants sont accessibles dans un réseau ambiant. Ces deux concepts sont intégrés dans la notion d'informatique ambiante. Dans cet ordre d'idées, nous allons définir les fonctionnalités de base d'une architecture de services ambiants pour faire interagir des dispositifs dans un environnement spontané et dynamique. A ce jour, il existe des architectures génériques pour gérer la communication entre entités informatiques [ITEA Ambience, 2004]. Parmi les architectures les plus exploitées, on peut citer : Jini (*Java intelligent network infrastructure*), UPnP (*Universal Plug and Play*), OSGi (*Open Services Gateway initiative*), CORBA (*Common Object Request Broker Architecture*), et Web Services. Nous allons analyser et effectuer une synthèse de ces architectures afin de les caractériser en fonction de nos critères de choix. Nous avons identifié que l'architecture UPnP est la plus adéquate et performante pour accomplir les objectifs de nos travaux. Finalement, nous allons proposer un modèle d'objet communicant en incorporant des fonctionnalités génériques inspirées par le concept de service ambiant.

3.1 Introduction aux architectures de services ambiants

3.1.1 La notion d'informatique ambiante

D'un point de vue technologique, on doit disposer d'un environnement informatique de gestion automatique et spontané de ressources afin de permettre aux acteurs de la chaîne logistique d'accéder à des services au travers d'un ou plusieurs réseaux d'accès filaire ou non. Pour cela, nous nous appuyons sur le concept d'informatique ambiante. Nous précisons que le concept d'informatique ambiante est associé au concept d'*ubiquitous computing* présenté dans le chapitre 2. Dans ce sens, les termes d'informatique ubiquitaire [Coutaz et al., 2002] [Oprescu, 2004] et d'informatique omniprésente [Gransart et al., 2001] [Jacquemin et al., 2005] sont souvent utilisés comme termes équivalents au terme anglophone *ubiquitous computing*.

L'informatique ambiante [Ciarletta, 2002] repose sur l'association des deux concepts clés que sont : la notion d'intelligence fournie par l'informatique, d'une façon générale, et la notion de présence ambiante, partout et toujours dans nos environnements, c'est-à-dire, d'ubiquité. Pour [Barralon et al., 2004] l'informatique ambiante est un moyen favorisant l'émergence opportuniste d'écosystèmes par couplage de ressources physiques. Dans ce sens, l'utilisateur construit à façon son espace d'interaction à la faveur des besoins et des ressources disponibles.

La Figure 35 illustre la notion d'informatique ambiante qui permet la communication en réseau entre multiples domaines d'interaction par des ressources physiques. La partie supérieure de la figure montre des domaines d'interaction de nature privée personnelle (voiture, banque, famille ...), de nature publique (transport, ...), et de nature organisationnelle (entreprise, ...). La partie inférieure montre les moyens techniques qu'un utilisateur peut employer pour accéder à l'information associée aux scénarios. Ces interfaces d'accès ou de médiation à l'objet peuvent être des téléphones portables, des ordinateurs personnels, des terminaux portables PDA, etc.

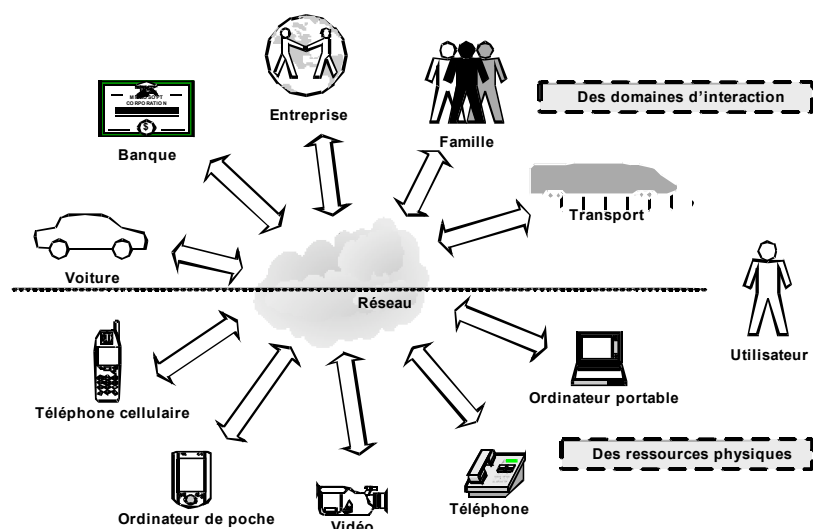


Figure 35 : Domaines et ressources du concept d'informatique ambiante.

Dans un environnement spontané [Duda, 2003], composé par des services et des ressources informatiques matérielles ou logicielles, les services disponibles s'adaptent dynamiquement à leurs contextes. Dans l'informatique ambiante, l'environnement informatisé doit avoir conscience de l'environnement dans lequel il travaille. Il doit percevoir les événements, la géographie, la localité, la présence des utilisateurs et l'évolution de ce contexte. Cette adaptabilité dans les interactions doit intégrer différents dispositifs connectés à une infrastructure des services réseau, communiquant et collaborant pour rendre de nouveaux services aux acteurs logistiques. Dans notre approche, le concept d'informatique ambiante représente l'ensemble de ressources informatiques disponibles, distribuées et non visibles, permettant d'offrir des services aux acteurs de la chaîne logistique. La notion d'informatique ambiante transférée au domaine de la production et de la chaîne logistique, implique que dans un futur très proche l'informatique et les technologies, d'une manière générale, vont devenir de plus en plus omniprésentes dans l'industrie et vont s'intégrer à l'environnement industriel de façon transparente pour les utilisateurs. De plus, les équipements industriels seront complètement interconnectés, capables de communiquer entre eux et de se configurer de manière dynamique et spontanée [Jammes et Smit, 2005].

3.1.2 Le concept de réseau ambiant

Un réseau ambiant couvre plusieurs aspects dans le monde de l'informatique, que l'on peut caractériser par des « sphères d'influences » [Ciarleta, 2002] où les ressources informatiques sont présentes dans différents espaces d'interaction et sont interconnectées avec d'autres ressources proches ou distantes. Les sphères d'influences s'organisent en partant de ce qui est le plus proche de nous jusqu'à la sphère la plus éloignée de notre environnement immédiat. Les sphères sont (Cf. Figure 36.) : la sphère individuelle (environnement d'interaction personnelle), la sphère de proximité (environnement immédiat pour communiquer avec de dispositifs proches), la sphère de l'organisation (environnement professionnel et industriel), la sphère globale (environnement virtuel pour la communication globale entre dispositifs). Dans ce sens, un réseau ambiant doit permettre de communiquer entre ressources informatiques, y compris des objets communicants, dans et entre toutes les sphères d'influences.

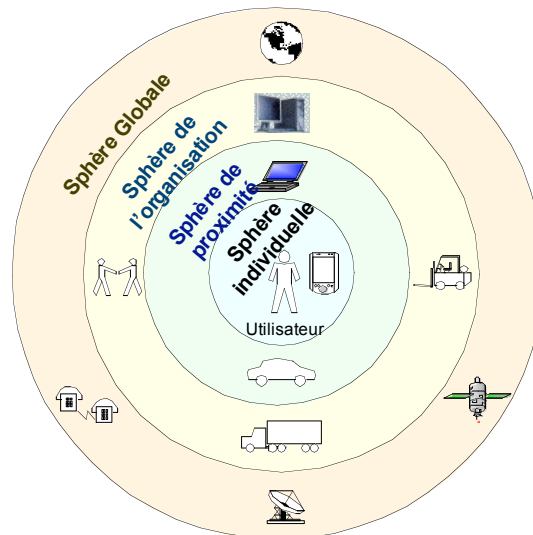


Figure 36 : Sphères d'influence dans un réseau ambiant.

D'après [Oprescu, 2004] un réseau ambiant est caractérisé pour fournir une connexion continue à des entités informatiques de moins en moins visibles et de plus en plus dynamiques. Pour [Rouseau et al., 2003] la notion de réseau ambiant représente une infrastructure assistant des applications et des dispositifs afin de fournir les fonctionnalités de base demandées par les applications telles que le stockage d'information, le traitement de l'information et la communication de l'information entre dispositifs. De plus, c'est un réseau capable de supporter, grâce à ses caractéristiques structurelles, un comportement coopératif, autonome, spontané, dynamique et sensible au contexte de toutes leurs entités participantes [Duda, 2003]. Dans le contexte du projet *Ambient Networks Project*³⁶, le concept de « *Ambient Network* » est associé à la capacité de communiquer des réseaux de communication hétérogènes afin de faciliter l'accès distant aux services offerts par différents opérateurs [Schieder, et al. 2004].

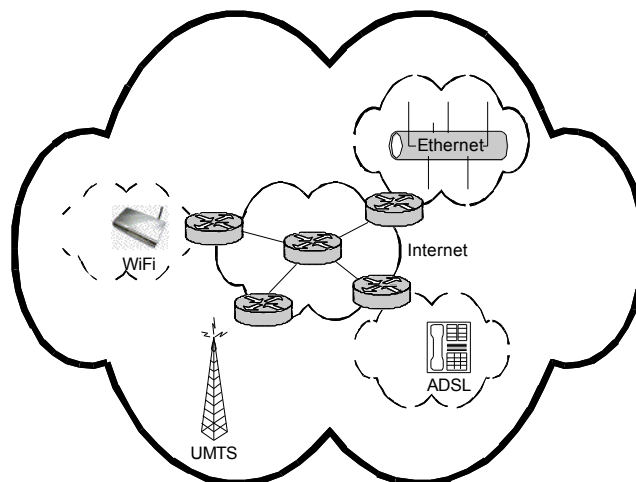


Figure 37 : Liaison entre différents types de réseaux ambiants

La Figure 37 représente un exemple de liaison entre différents types de réseaux ambiants en incorporant différentes technologies de communications telles que WiFi, Ethernet, UMTS, ADSL liées par des technologies Internet. Ces technologies sont caractérisées pour offrir des niveaux de qualité de services non uniformes en termes de couverture géographique, débit, délai, taux d'erreurs et sécurité.

³⁶ www.ambient-networks.org - *Ambient Networks Project*

Dans ce cas, le but pour un dispositif mobile est d'être capable de choisir dynamiquement la meilleure option de communication en fonction de ses besoins.

Nous reprenons ici le concept de « *Ubiquitous Computing* », proposé par [Weiser, 1991], qui vise à connecter tout type de dispositif en tout lieu et à tout moment. En ce sens, notre vision est de relier les objets communicants avec les autres acteurs de la chaîne logistique à tout moment de son cycle de vie et en tout lieu géographique. Pour atteindre cet objectif, il est indispensable d'assigner des fonctionnalités d'auto configuration à des dispositifs qui soient reliés à un réseau ambiant. Il faut que les dispositifs soient organisés pour fonctionner de manière autonome. Par exemple, un dispositif pourra découvrir les ressources et les produits avec ses services disponibles sur un réseau industriel, les charger, ou les initialiser pour un usage à distance, selon le type d'application exécutée par l'acteur de la chaîne logistique. Ces services doivent constituer un ensemble de fonctionnalités non intrusives et accessibles de manière intuitive.

3.1.3 Services Ambiants

Nous nous appuyons sur le concept de service ambiant afin de développer des interactions entre un objet communicant et les autres acteurs de la chaîne logistique. Ainsi, dans notre approche, le concept de service ambiant représente une vue abstraite pour définir des services de haut niveau accessibles par un réseau ambiant, permettant de fournir des possibilités de gestion de l'information, des possibilités de traitement de l'information et de gestion d'événements [Cea et Bajic, 2004]. Pour [Rousseau et al., 2003] un service ambiant est un module de communication capable de traiter l'information, fournir une interface de contrôle et publier la description de service. Selon [Naing et al., 2003] les services ambiants peuvent être considérés selon une caractéristique géolocale, dénommés ainsi *location-based services*, c'est-à-dire, des services déterminés selon la localisation géographique de l'équipement. Cette notion est adoptée aussi par [Loke et al., 2005], pour délimiter un service ambiant par une frontière géographique de pertinence et d'utilité pour les utilisateurs. Aussi, les auteurs proposent de considérer des classifications de services par rôle, par type d'activité, ...) dans la frontière géographique afin d'organiser les services à offrir aux utilisateurs selon chaque situation.

Dans un contexte de services ambiants, le produit peut être un fournisseur ou un demandeur de services, en exigeant des mécanismes de découverte et de contrôle des services offerts dans un domaine de services déterminé. Dans notre cas, un domaine peut être de caractère fonctionnel ou géographique. Ce dernier cas peut être assimilé à un domaine de travail. Dans le cas d'un domaine de service fonctionnel, les services disponibles sont groupés par affinité en fonction d'un ou plusieurs critères donnés. En revanche, dans le cas d'un domaine de service géographique, les services offerts sont groupés en considérant leur localisation à l'intérieur d'un système. Dans [Loke et al., 2005] un domaine de services est représenté par un ensemble de services associés à une zone géographique déterminée. La Figure 38 illustre ces concepts autour de trois domaines de services. Le domaine de services A représente un atelier de production et le domaine de services B représente un entrepôt de stockage de produits : ces deux domaines sont de caractère géographique. Le domaine A' est un domaine fonctionnel représentant la fonction de contrôle d'opérations liées aux deux domaines géographiques.

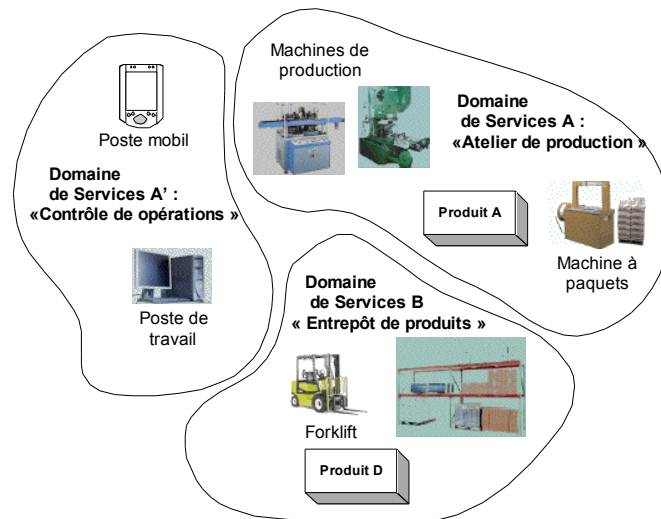


Figure 38 : Domaines de services dans un réseau ambiant.

Un produit communicant inséré dans un réseau ambiant peut apparaître ou disparaître dans un domaine de service au gré de sa circulation et de ses mouvements physiques. Ce comportement représente l'aspect sponténé et dynamique d'un produit, selon sa nomadicité dans son cycle de vie. L'accés au produit / objet communicant dans un domaine de travail exige, d'abord, son identification automatique et, ultérieurement, la configuration spontanée des n&ocedil;uds de communication et de leurs services associés dans le réseau ambiant.

3.1.4 Paradigmes de communication

Dans les architectures de systémes informatiques distribués, il existe deux paradigmes de communication dominants afin de faire interagir les entités [Schlueter, 2003] : le paradigme client / serveur et le paradigme des systémes événementiels (*event - based system*). D'une part, le paradigme client / serveur considére deux types d'entités élémentaires : les clients et les serveurs [Ruffer, 1995]. Un client représente une entité qui initie une action en invoquant une requ&eate vers un serveur distant avec une finalité déterminée. Autrement dit, un client est un demandeur de services. Un serveur représente une entité qui a pour r&ocedil;le de répondre aux requ&eates réalisées par les clients. C'est-à-dire, le serveur est un fournisseur de services. La distinction entre client et serveur est seulement conceptuelle, il est possible qu'une m&eame entité soit un client et un serveur. D'autre part, le paradigme des systémes événementiels représente un systéme composé d'entités capables d'envoyer et recevoir des messages d'événements survenus dans le systéme, et non sollicités. Le principe de «souscription / notification» était ce paradigme [Huang et Garcia-Molina, 2004], ainsi une entité cliente s'abonne aux événements aupr&e;s d'une autre entité serveur. Lorsqu'une entité serveur met à jour son état interne ou sa relation avec d'autres entités, elle publie ces événements aux entités clientes souscripteurs.

Selon [Briscoe, 2004], le mod&e;le de type événementiel « *subscriber / publisher* » sera le mod&e;le de communication prédominant dans les interactions « machine to machine » (M2M) [France Télécom, 2006] [Wang et al., 2006]. La Figure 39 illustre les interactions entre les entités en fonction du paradigme adopté. La partie supérieure de la figure montre un Client A exécutant une action sur le Serveur B. Ensuite, le serveur B lui répond. La partie inférieure de la figure montre un Demandeur A qui souscrit aux événements du Fournisseur B chez un gestionnaire d'événements. Lorsqu'un événement se produit sur le Fournisseur B, ceci est notifié au gestionnaire d'événements. Ce dernier

informe de l'événement en notifiant le Demandeur A. Nous remarquons que le modèle de type événementiel permet d'implémenter dans le client un comportement réactif.

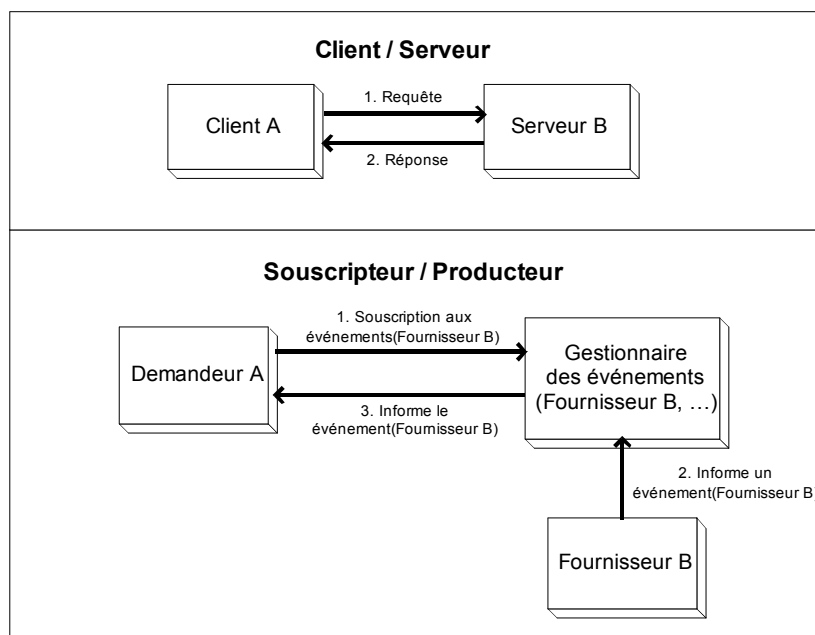


Figure 39 : Représentation du paradigme client/serveur et du paradigme souscripteur/producteur

3.1.4.1 Paradigme de communication appliqué au domaine de la chaîne logistique

Dans la chaîne logistique, nous rencontrons de multiples acteurs. Un acteur de la chaîne logistique est une entité qui peut être un demandeur de services ou un fournisseur de services. Classiquement, comme demandeurs de services, nous pouvons signaler les utilisateurs humains (par exemple des opérateurs), les ressources physiques (par exemple des machines, des robots, etc.) et les ressources informatiques (par exemple une application informatique de prestataires). Nous proposons comme nouveau demandeur de services un produit / objet communicant (ou produit intelligent). Dans ce cas, le produit communicant représente un produit physique enrichi de capacités de mémorisation, communication, perception, action et décision. D'ailleurs, nous trouvons habituellement dans la chaîne logistique des utilisateurs, des ressources physiques et des ressources informatiques étant des fournisseurs de services.

Dans les dernières années, les produits physiques ont pris le rôle de fournisseurs d'information grâce à des données portées par eux qui peuvent être transformés facilement en information. Depuis un objet communicant a été vu comme un fournisseur de services par [Kärkkäinen et al., 2003] (notion de produit / objet communicant), [Rieki et al., 2005] (notion de produit / objet communicant) et comme un demandeur / fournisseur de services par [Privat, 2000] (notion de dispositif), [Smirnov et al., 2004] (notion de ressource informatique), [Jarvis et al., 2006] (notion de holon). Nous proposons de considérer un produit / objet communicant comme un fournisseur et un demandeur de services [Cea et Bajic, 2004].

De façon générale, un acteur ou processus de la chaîne logistique peut être considéré comme un fournisseur ou un demandeur de services. La Figure 40 montre le mode de communication générique associé au paradigme « client/serveur » et illustre la dualité demandeur / fournisseur de services d'un produit communicant. L'acteur de la chaîne logistique réalise une requête par annonce d'une demande

de service vers le produit communicant, ou bien, il répond à la requête faite par le produit communicant.

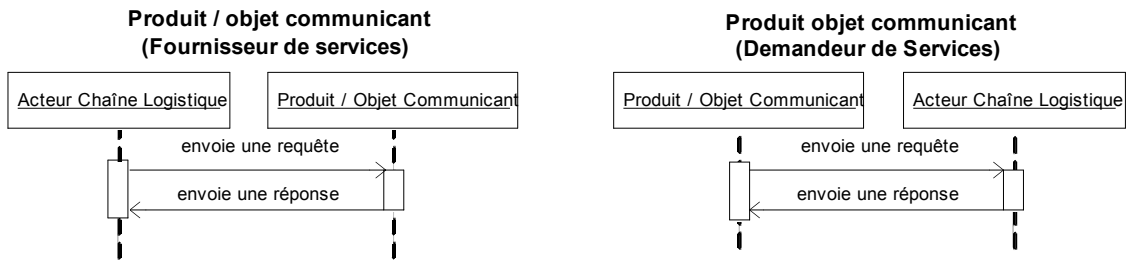


Figure 40 : Mode de communication « request / reply » : le produit / objet communicant comme fournisseur et demandeur de services

Par rapport au paradigme événementiel, nous citons le travail de [Cheverst et al., 2002] qui propose un système de gestion d'information contextuelle dans lequel les changements d'information présentés aux utilisateurs sont déclenchés par des changements dans la localisation physique de l'utilisateur. Dans cet esprit, un acteur de la chaîne logistique étant un demandeur de service, peut s'abonner auprès d'un fournisseur de services afin de s'informer de tout changement concernant l'état du fournisseur. Dans ce cas, un produit communicant a besoin de disposer d'information actualisée pour prendre les décisions adéquates. Vice-versa, le produit communicant peut informer des changements d'état à tous les demandeurs de services intéressés. La Figure 41 montre le mode de communication générique associé à un produit communicant en considérant le paradigme de « système événementiel ». Dans la Figure 41 on observe que les fournisseurs de services sont les gestionnaires des demandes de souscription. Ainsi, le produit communicant souscrit aux événements de l'acteur de la chaîne logistique. Lorsque ce dernier réalise une opération concernant la souscription faite par le produit, l'acteur de la chaîne logistique l'informe ensuite ce changement. Ensuite le produit peut évaluer l'impact de ce changement sur lui-même. On observe aussi que l'acteur fait une souscription pour recevoir des événements associés au produit communicant. Dans ce cas, après que le produit a réalisé une opération concernant une souscription de l'acteur, il l'informe de cette nouvelle situation.

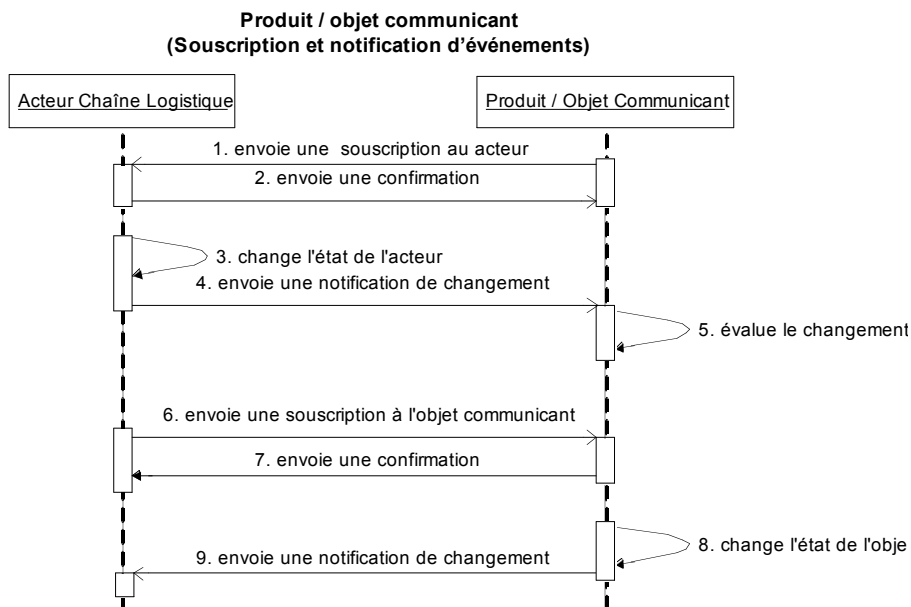


Figure 41 : Mode de communication « souscription / notification » : le produit intelligent comme annonceur et souscripteur d'événements.

Dans notre approche, les interactions entre acteurs sont supportées par le concept de service. Ce principe a été aussi adopté par [Jarvis et *al.*, 2006] dans le domaine des systèmes de production holoniques. En effet, il y aura des interactions quand les demandeurs et fournisseurs de services sont en communication. Les interactions entre les acteurs dans les différentes phases de son cycle de vie, requièrent divers services génériques ou spécifiques en fonction de leurs besoins. Comme élément implicite de l'interaction entre acteurs, nous identifions le concept de contexte. Le contexte comme élément appliqué [Dey, 2000] est représenté par les caractéristiques des ressources informatiques employées pour effectuer les interactions entre les produits communicants et les acteurs de la chaîne logistique, les caractéristiques des utilisateurs et des ressources physiques qui sont en relation avec les objets communicants et les caractéristiques de l'environnement physique. Il est évident que le contexte influence le comportement de l'objet communicant et des acteurs de la chaîne logistique.

3.2 Caractéristiques d'une architecture de services ambiants

Cette section explore les caractéristiques nécessaires pour mettre en œuvre le concept de service ambiant. Pour cela, il faut définir et caractériser une architecture de services ambiants. [Langheinrich et *al.*, 2000] a proposé la nécessité d'une infrastructure pour des « *smart things* » puisque les objets peuvent avoir besoin d'accéder à l'information sur sa localisation, de découvrir son environnement immédiat et peuvent vouloir communiquer avec d'autres objets.

Dans cette partie, nous allons définir les fonctionnalités génériques d'une architecture de gestion de services ambiants composée par des dispositifs. Notre approche consiste à considérer les acteurs de la chaîne logistique comme des dispositifs qui peuvent demander et/ou fournir des services. Comme nous l'avons dit, dans notre approche, un produit physique enrichi avec des capacités de mémorisation, de perception, de communication, d'action et de décision peut devenir un produit communicant. Ce dernier, peut être représenté, globalement, comme un dispositif en communication avec un ensemble de dispositifs mobiles ou fixes. Cette distinction sera clarifiée à la fin de ce chapitre au travers des fonctionnalités spécifiques que nous proposons pour un produit communicant.

3.2.1 L'identification automatique d'un dispositif

Cette fonctionnalité permettra d'identifier automatiquement de façon unique un dispositif ou ressource physique dans un réseau ambiant. Puisque les dispositifs peuvent apparaître ou disparaître dans le réseau ambiant, il faut disposer de mécanismes qui permettent de mettre à jour le nombre de dispositifs présents effectivement dans le réseau. Chaque dispositif doit automatiquement être identifié, sans aucune intervention humaine ou administrative, comme élément constitutif d'un réseau ambiant. L'identification précise d'un dispositif requiert d'assigner à chaque dispositif un identifiant unique [Römer et *al.*, 2004]. Cette identification peut reposer notamment sur des technologies RFID, des technologies d'identification biométrique [Fuentelba, 2005] ou des technologies d'identification par signature informatique.

3.2.2 La localisation d'un dispositif

Il est nécessaire de disposer d'une fonctionnalité de recherche adaptée pour savoir où se trouve un dispositif physique dans un réseau ambiant. La localisation géographique d'un dispositif est importante dans le but d'adapter les services au domaine de travail ou simplement comme moyen de repérage topologique. Ainsi, cette fonctionnalité permet d'associer une référence spatiale – temporelle

à un dispositif afin de configurer l'accès aux services en fonction du contexte du demandeur de services (dispositif). En fait, les dispositifs peuvent augmenter la connaissance du contexte en connaissant sa localisation [Mattern et Sturm, 2003]. Le dispositif peut s'informer sur sa localisation géographique et sur les conditions environnementales, par exemple, au moyen de capteurs intégrés ou non, afin de réagir ou s'adapter à l'environnement dans lequel il est situé.

3.2.3 La découverte automatique des services

Cette fonctionnalité permet la découverte automatique et spontanée des services associés à un dispositif par d'autres éléments dans le réseau ambiant. En utilisant des mécanismes de communication avancés, il est nécessaire qu'un demandeur de services puisse localiser les services disponibles et récupérer ses paramètres d'appel, et qu'un fournisseur de services puisse annoncer ses services disponibles aux entités connectées à un réseau ambiant. Ce premier mécanisme de recherche peut être représenté comme le processus de comparaison entre une requête (description partielle du service souhaité qui est formulé par un client) et la description complète des services disponibles obtenus des serveurs. En effet, une fois qu'un fournisseur de service arrive à un réseau ambiant, ses services doivent être publiés automatiquement. Si un fournisseur de service disparaît, cette situation doit être connue pour tous les dispositifs connectés au réseau ambiant. Une classification des mécanismes de découverte des services a été proposée par [Zhu et al., 2002], en soulignant l'existence de deux modes permettant de connaître les services disponibles à un moment donné : le mode centralisé et le mode distribué.

3.2.3.1 Mode centralisé

Le mode centralisé, repose sur un répertoire central, ou un catalogue, qui contient tous les services publiés par les serveurs de services [Zhu et al., 2002] [Duda, 2003] [Zhu et al., 2005]. Dans cette approche de découverte centralisée des services, la découverte peut être passive ou active. Une découverte sera passive si la communication est initiée par le serveur et une découverte sera active si la communication est initiée par le client. La découverte passive, ou mode de découverte « *push* », considère que le répertoire centralisé émet des avertissements aux clients pour informer les services disponibles dans le catalogue centralisé. En revanche, dans une découverte active ou « *pull* », le client de services fait des requêtes décrivant le type de service désiré auprès d'un répertoire centralisé, et ensuite le répertoire lui répond avec les services demandés. En effet, avant la découverte des services, il y a une étape nécessaire d'enregistrement des services auprès du répertoire central de services.

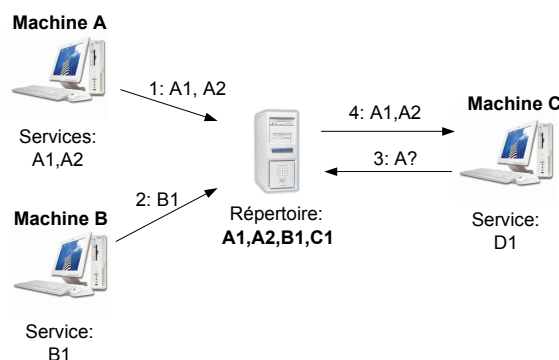


Figure 42 : Enregistrement et découverte active des services en mode centralisé.

La Figure 42 représente l'enregistrement et la découverte des services d'un dispositif dans un réseau ambiant en mode centralisé. La Machine A enregistre ses services A1 et A2 sur le répertoire centralisé. Ensuite, la machine B informe son service B1 sur le serveur centralisé. Après, la machine C demande au répertoire centralisé s'il a des services de type A, en obtenant comme réponse les services A1 et A2 offerts par la Machine A.

3.2.3.2 Mode distribué

Le mode distribué représente un mécanisme de recherche décentralisé. Dans ce deuxième cas, chaque serveur a la responsabilité de publier ses services sur le réseau ambiant [Zhu et al., 2002] [Zhu et al., 2005]. Dans une architecture distribuée, la découverte peut être de type aussi « *push* » ou « *pull* ». Dans le mode « *push* » ou mode passif, les serveurs annoncent les services dont ils mettent à disposition des clients au travers d'un mécanisme d'avertissement de services. Donc les clients écoutent ces annonces. Dans le mode « *pull* » en incorporant une découverte active, les serveurs répondent avec les services à la requête de services des clients. Si le client demande un service non disponible, il ne reçoit aucune réponse.

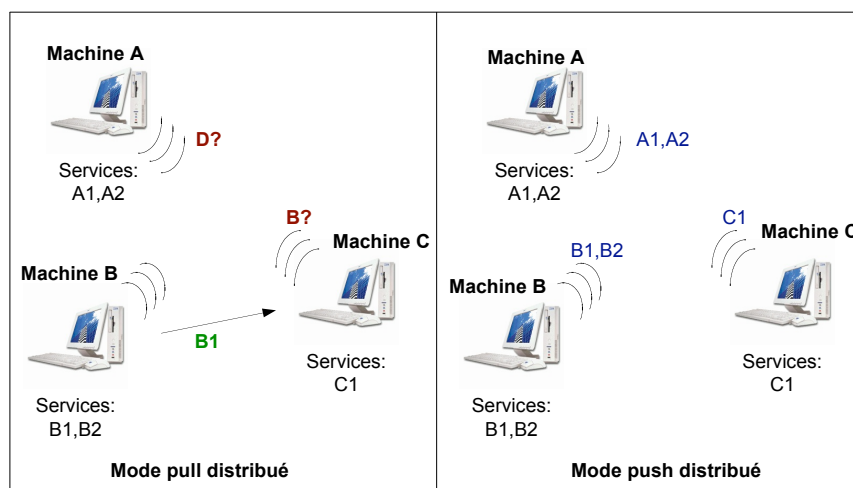


Figure 43 : Découverte des services en mode distribué.

La Figure 43 représente le mode distribué afin de découvrir les services disponibles dans un réseau ambiant. La partie gauche de la figure montre le mode *pull* distribué. On observe que la Machine B répond avec le service B1 à la requête de la Machine C qui est intéressée par un service de type B. La Machine A demande un service de type D, et elle ne reçoit aucune réponse parce qu'un tel service n'est pas disponible. La partie droite de la figure montre le mode *push* distribué. Ici, les Machines A, B et C annoncent les services dont elles disposent. Si une machine quelconque est intéressée par les services offerts, elle pourra contacter la machine qui correspond.

3.2.3.3 Mode centralisé versus mode distribué

Le mode centralisé requiert une infrastructure dédiée pour maintenir le répertoire centralisé. Par contre, dans le mode décentralisé, il n'est pas requis de répertoire, en supprimant les risques associés aux défaillances du répertoire centralisé pouvant engendrer l'interruption du service de découverte. Toutefois, une architecture distribuée requiert une plus grande consommation d'énergie due au fait que les fournisseurs de services opèrent et gèrent leurs services de manière autonome continuellement.

D'autre part, dans une architecture distribuée, les temps de réponse sont plus petits [Mathieu *et al.*, 2001] étant donné l'existence d'un plus grand nombre de serveurs disponibles pour répondre aux clients et parce que les serveurs ont une plus petite charge de travail par rapport à une situation centralisée avec un seul serveur centralisé. En ce qui concerne la complexité des requêtes, dans une approche centralisée, en disposant d'un répertoire central avec la description de tous les services enregistrés, il est possible de répondre à des demandes complexes du client. Ce processus requiert d'exécuter des mécanismes de comparaison entre les différents services enregistrés et disponibles dans l'environnement, et ensuite, d'informer le client de la meilleure option. Par contre, dans une approche distribuée, il n'est pas possible de faire ce processus de comparaison du côté du serveur parce qu'il n'a pas toutes les informations nécessaires, mais il est tout à fait possible de mettre en place des mécanismes de comparaison du côté des clients pour améliorer la performance du processus de découverte. Ainsi, la quantité de messages échangés entre des entités informatiques est, en général, plus grande dans une architecture distribuée que dans une architecture centralisée [Oprescu, 2004], en impliquant de disposer d'une bande passante plus grande pour implémenter une approche distribuée par rapport à une approche centralisée dans des situations semblables [Govea et Barbeau, 2000].

Il est possible de combiner le mode « *pull* », essentiellement réactif, avec le mode « *push* », plutôt proactif, pour avoir un mode mixte de découverte des services. Le mode mixte combine les deux modes indiqués, en cherchant à bénéficier des avantages de chacun en minimisant les points faibles individuels. De cette façon, les clients font des requêtes lorsqu'ils ont besoin d'un service et, parallèlement, ils reçoivent des annonces de services fournis par les serveurs.

3.2.4 La description d'un dispositif et de ses services

Après la découverte d'un dispositif et de ses services, il est nécessaire d'obtenir plus d'information sur la structure physique et logique d'un dispositif, ainsi que le détail de chaque service découvert. Par conséquent, l'objectif de cette fonctionnalité est de spécifier et de décrire la structure d'un dispositif et de tous ses services internes associés. Cette spécification doit être standardisée afin de permettre l'interopérabilité entre les différents fournisseurs des services. La Figure 44 illustre le mécanisme de description de dispositifs et de ses services basé sur des documents standardisés interprétables par des demandeurs de services qui rendent compte de leur présence.

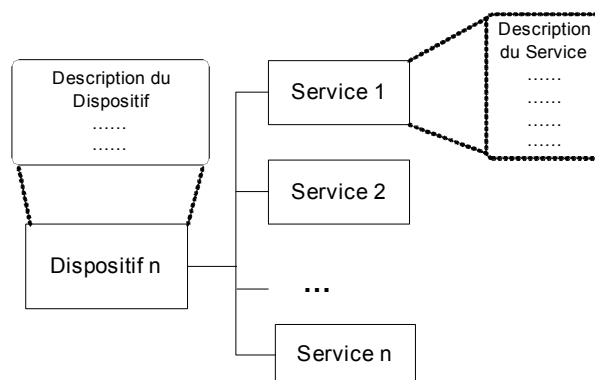


Figure 44 : Description standardisée d'un dispositif et de ses services.

Comme ligne de travail intéressante dans ce domaine, nous citons la proposition de [O'Sullivan, 2002] afin de compléter la description fonctionnelle d'un service avec des paramètres non fonctionnels tels que la qualité de service, la sécurité du service, le propriétaire du service, la consommation du service, le coût du service, les droits du demandeur de services. L'incorporation de ce type

d'information complémentaire permettrait d'affiner la recherche d'un service dans un réseau ambiant et de prendre en compte plus d'informations au moment où un demandeur de services prend des décisions.

3.2.5 L'invocation des services

L'objectif de cette fonctionnalité est de pouvoir mettre en place des mécanismes pour faire interagir les entités, conformant le réseau ambiant, avec leurs services offerts en recherchant une communication dynamique, effective et coopérative. Lorsqu'un demandeur de service est intéressé par un service particulier, le demandeur de service ou client, exécute le processus d'invocation de services vers les fournisseurs de services. Ces actions à distance permettent de contrôler un dispositif et d'invoquer leurs services associés. Les services peuvent être surveillés ou contrôlés par tous les dispositifs autorisés qui font partie du réseau ambiant. L'accès à un espace personnalisé de services est conditionné par le profil des clients [Bilchev et al., 2005]. Ce profil est associé à une activité particulière développée par un demandeur de services dans une localisation déterminée à un moment donné [Figge, 2004], c'est-à-dire, le profil est associé à un domaine de services. L'invocation de services peut se produire seulement si les services sont actifs du côté du fournisseur. Le départ ou l'arrivée d'un service au réseau ambiant est associé au mécanisme de découverte de services. L'information de ce fait doit être donnée aux demandeurs des services.

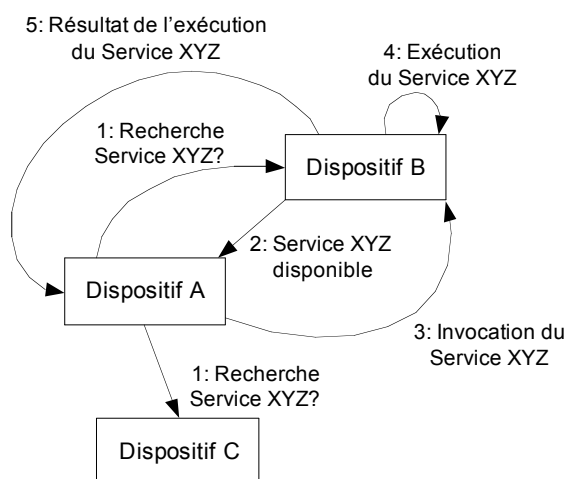


Figure 45 : Diagramme d'échange pour l'invocation de service en mode distribué.

La Figure 45 montre un diagramme indiquant la séquence de messages pour rechercher un service déterminé en mode pull distribué et l'invoquer à distance. Le Dispositif A recherche le service XYZ en faisant une demande à tous les fournisseurs de services présents dans le réseau ambiant. Le Dispositif B l'informe qu'il a le service demandé. Ensuite, le Dispositif A invoque le service sur le Dispositif B. Après avoir exécuté le service XYZ, le Dispositif B informe le Dispositif A de son résultat.

3.2.6 La notification des événements

La notion de notification d'événements permet d'informer opportunément les entités intéressées lorsqu'un changement se produit dans l'état des services d'un dispositif. Ce mécanisme de notification est basé sur le mode de communication « publish / subscribe » [Briscoe, 2004]. Cette fonctionnalité

requiert, premièrement, que les clients connaissent les services disponibles offerts par les fournisseurs de services, et décident de faire ensuite, une souscription auprès de ces fournisseurs pour s'informer sur les nouveaux changements associés aux variables d'un service. Ainsi, les clients seront informés de tout changement qui pourrait les intéresser. Il est possible de définir des services basés sur ce mécanisme de notification comme par exemple dans le cas d'un service qui permet d'alerter un client quand une variable atteint une valeur critique.

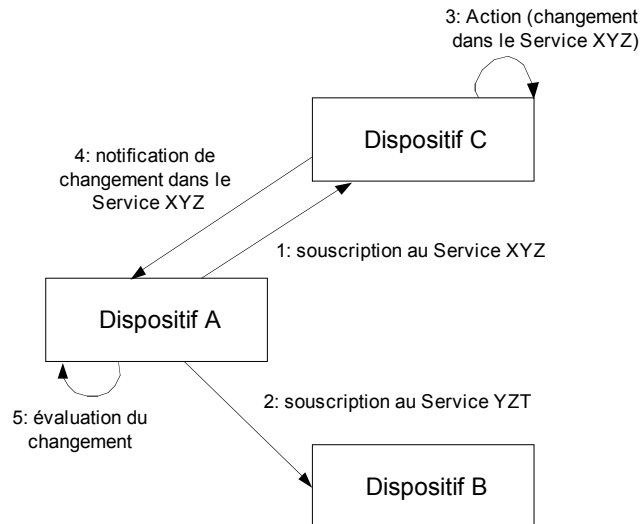


Figure 46 : Diagramme d'échange de souscription et notification vers un dispositif.

Le diagramme représenté dans la Figure 46 montre la séquence de messages entre les dispositifs afin de souscrire et notifier les événements générés par les fournisseurs de services. Le Dispositif A souscrit aux changements du Service YZT sur le Dispositif B et aux changements du Service XYZ sur le Dispositif C. Lorsque un changement dans le service XYZ se produit sur le Dispositif C, le Dispositif A est notifié. Finalement, le Dispositif A évalue l'impact de ce changement sur son comportement.

3.3 Architectures de gestion de services ambiants

Il existe plusieurs initiatives dans le monde de la recherche ainsi que dans le monde industriel afin de supporter la gestion de services ambiants des dispositifs dans un environnement informatique [Opreescu, 2004] [Zhu et al., 2005] [Marin-Perianu et al., 2005] [Vanthournout et al., 2005].

L'objectif d'une architecture de services ambiants est de découvrir automatiquement les dispositifs et leurs services, de décrire les services offerts par un dispositif, d'annoncer la disponibilité des services d'un dispositif et d'invoquer à distance les services d'un dispositif dans un réseau ambiant. Pour faire interagir les dispositifs dans un environnement donné, on dispose actuellement d'architectures très reconnues par la communauté scientifique et industrielle. Dans ce sens, nous allons analyser les principales architectures de services utilisées à ce jour : Jini (*Java intelligent network infrastructure*), UPnP (*Universal Plug and Play*), OSGi (*Open Services Gateway Initiative*), CORBA (*Common Object Request Broker Architecture*) et les Services Web (*Web Services*). Après la présentation et la description de chaque architecture, nous allons présenter une étude comparative afin d'identifier l'architecture la plus adaptée aux objectifs de cette thèse.

3.3.1 Jini (*Java intelligent network infrastructure*)

Jini est une architecture distribuée de services proposée par Sun Microsystems³⁷ qui s'appuie sur un environnement Java [Sun Microsystems, 1999]. Jini utilise le concept de fédération pour s'adresser à un ensemble de dispositifs autonomes coopérant. Les clients et les serveurs de services qui appartiennent à la communauté Jini doivent rechercher un répertoire de services avant de commencer à interagir. Jini permet l'existence d'un ou plusieurs répertoires de services (Jini Lookup Service) qui représente l'endroit où les fournisseurs déclarent leurs services. Les répertoires peuvent être recherchés activement par les clients et les fournisseurs de services par messages multicast. Une fois qu'un répertoire a été découvert, les clients et serveurs commencent l'association à la fédération. Après ce processus, le client a la possibilité de découvrir les services proposés dans la fédération, pendant que les serveurs peuvent enregistrer ses services. Dans l'architecture Jini, les services sont des objets Java et sont décrits dans une classe en encapsulant un objet, son identificateur, et ses attributs. Le processus de découverte de répertoires de services utilise le protocole UDP (*User Datagram Protocol*) et le reste des communications entre les participants à la communauté sont faits au moyen de procédure RMI (*Remote Method Invocation*). Ce dernier point implique une mobilité du code provoquant une consommation non négligeable de la bande passante. Lorsqu'un client décide d'utiliser un service découvert dans un répertoire, il récupère le *proxy* de l'objet et donc peut l'utiliser en invoquant à distance, les méthodes exportées par le service. Jini ne propose pas de mécanisme pour réaliser des filtrages des serveurs adaptés à la requête du côté du répertoire. Jini permet de faire des requêtes en s'appuyant sur l'égalité entre les valeurs des attributs recherchés et les valeurs du service. C'est donc le client qui doit filtrer les services selon ses attributs. Dans Jini, les enregistrements des services ont des temps de vie limités. Ceci est défini par le fournisseur du service. Un service est éliminé automatiquement du répertoire à l'expiration si le fournisseur ne renouvelle pas son service. En plus, Jini offre un mécanisme de notification d'événements distribués. De cette façon, lors d'apparitions, de disparitions ou de modifications des services, le répertoire avertit les clients intéressés.

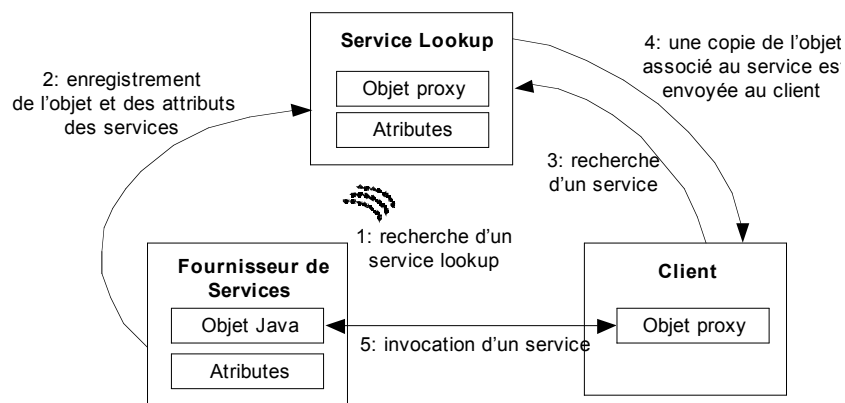


Figure 47 : Fonctionnement générale de Jini.

Le fonctionnement général de Jini est montré dans la Figure 47:

1. Un fournisseur de services recherche un *Service Lookup* par multicast afin d'enregistrer ses services.
2. Le fournisseur de services enregistre l'objet Java et les attributs de ses services dans le *Service Lookup*.

³⁷ www.sun.com/software/jini - Jini Network Technology

3. Un client cherchant un service déterminé, interroge le Service *Lookup* trouvé par le mécanisme de découverte.
4. Ce dernier lui retourne l'objet *proxy*. Le client télécharge donc automatiquement le code correspondant à cet objet s'il ne le possède pas.
5. Par l'invocation de la méthode en question dans l'objet *proxy* la communication entre le client et le service est réalisée.

Théoriquement Jini est seulement lié à la méthode RMI et aux protocoles Jini, mais par nature Jini est lié au langage Java en impliquant une architecture dépendant du langage de programmation [Melchor, 2004]. Des applications concrètes ont montré l'applicabilité de cette architecture, par exemple en créant une « *smart kitchen* » [Urnes et al., 2000] ou un « *smart map* » [Urnes et al., 2001] chez Telenor en Norvège. Dans le contexte d'une application expérimentale, [Römer, 2004] envisage des problèmes de gestion d'embouteillage avec le traitement d'une grande quantité d'entités en accédant à un répertoire de services centralisé Jini.

3.3.2 UPnP (*Universal Plug and Play*)

L'architecture UPnP est une architecture réseau ouverte et distribuée, soutenue initialement par Microsoft, et supportée par le forum UPnP³⁸. Le forum est composé par plus de 750 sociétés y compris des leaders dans les secteurs de l'électronique, l'informatique, la domotique, les dispositifs réseaux, les appareils mobiles, les imprimantes, etc. Le forum UPnP cherche à développer des standards pour permettre une connectivité simple et robuste entre des appareils indépendants quels que soient les constructeurs et les ordinateurs. Dans UPnP il y a deux entités principales : le **dispositif** et le **point de contrôle**. Un dispositif est un appareil qui offre des services, et un point de contrôle est l'entité qui les découvre et les utilise [UPnP Forum, 2003]. Un dispositif peut contenir d'autres dispositifs embarqués. Un service contient des actions avec ses arguments d'entrée et sortie, et une liste des variables dont les changements de valeurs peuvent être notifiés aux clients intéressés.

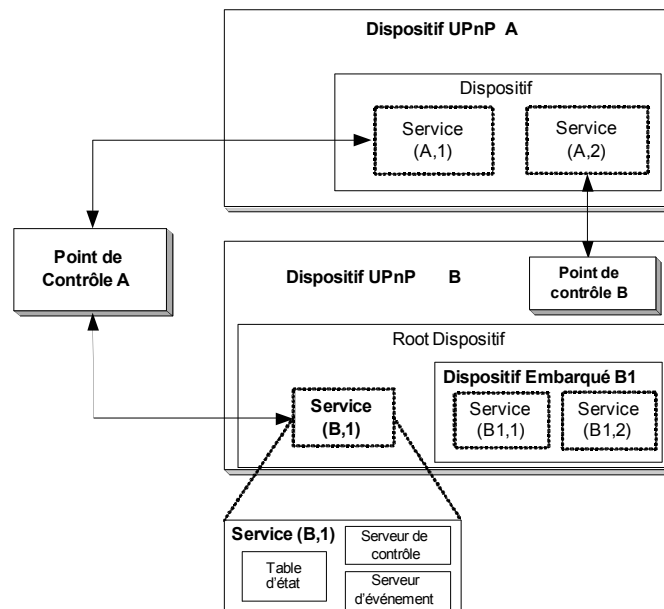


Figure 48 : Points de contrôle UPnP en invoquant des services vers des dispositifs UPnP

³⁸ www.upnp.org - Universal Plug and Play Forum

La Figure 48 montre un Point de contrôle A demandant le service (A,1) au Dispositif A et le service (B,1) au Dispositif B. Dans le même temps, un Point de contrôle B, intégré dans le Dispositif B, demande le service (A,2) au Dispositif A. On observe également que le Dispositif B dispose d'un Dispositif B1 embarqué contenant les services (B1,1) et (B1,2).

Lorsqu'un dispositif ou point de contrôle rejoint le réseau, il obtient automatiquement une adresse, soit par Autoaffectation IP, soit par mécanisme DHCP (*Dynamic Host Configuration Protocol*). Pour découvrir l'environnement, le protocole utilisé par UPnP est le protocole SSDP (*Simple Service Discovery Protocol*). Lorsqu'un dispositif, ou un point de contrôle, vient de se connecter au réseau, il annonce sa présence par message multicast en utilisant les protocoles SSDP et HTTP multicast sur UDP (*User Datagram Protocol*). Le message d'annonce contient un URI (*Universal Resource Identifier*) et l'URL d'un document de description du dispositif et les services offerts. Cette description est exprimée en format XML (*eXtended Mark-up Language*). Un point de contrôle a la possibilité de découvrir des dispositifs et des services en multicast à l'aide de SSDP et HTTP sur UDP. Un point de contrôle, après la découverte, peut obtenir plus d'informations en récupérant le fichier XML qui contient une description du dispositif, la liste des services offerts en incorporant l'URL de la description détaillée du service, l'URL pour contrôler le service et l'URL pour souscrire aux événements. Un point de contrôle utilise la liste des actions et l'URL de contrôle pour invoquer des actions à distance par le protocole SOAP (*Simple Object Access Protocol*), ne nécessitant pas la mobilité de code entre des applications. Si le point de contrôle est abonné aux événements auprès des services, il pourra recevoir des notifications lors de changement d'une variable d'état du dispositif. Le protocole de notification employé par UPnP est le protocole GENA (*Generic Event Notification Architecture*). L'URL de présentation informée dans la description du dispositif, permet aux points de contrôles de visualiser l'état d'un dispositif et de contrôler son fonctionnement à distance. Ainsi, UPnP exige que les interactions entre des dispositifs soient effectuées en employant des protocoles standardisés.

En résumé, UPnP est une architecture distribuée caractérisée pour gérer automatiquement la connexion et la découverte des dispositifs et leurs services sur un réseau ambiant IP (*Internet Protocol*). Avec UPnP, un dispositif ou un point de contrôle peut joindre dynamiquement un réseau ambiant, obtenir automatiquement une adresse IP, distribuer ses services et connaître la présence des services des autres dispositifs [Jeronimo et al., 2003].

En France, le Groupe des Ecoles de Télécommunications (GET), associant l'INT d'Evry, l'ENST Bretagne de Brest, et l'ENST de Paris, a choisi l'architecture UPnP pour développer des applications dans le domaine de la « maison intelligente » avec le but de permettre la découverte d'équipements de la maison et l'envoi de commandes dédiées [Segarra et al., 2003] [Ghorbel et al., 2004]. Comme résultat de ce projet le système AMETSA a été créé en proposant un point de contrôle générique permettant le contrôle de tous les dispositifs raccordés au réseau UPnP dans une maison. L'implémentation a été faite sur Linux en utilisant des composants de base proposés par Intel³⁹. Il est important de signaler que des applications basées sur les protocoles UPnP peuvent être compatibles avec un environnement de travail OSGi [Smit, 2004], permettant de cette façon une interopérabilité entre des dispositifs UPnP et des composants OSGi. Dans le domaine industriel, l'architecture de services proposée par le projet ITEA SIRENA (*Service Infrastructure for Real-Time Embedded Network Applications*), a été basée sur l'architecture UPnP.

³⁹ www.intel.com/technology/upnp - Intel UPnP Technology

3.3.3 OSGi (*Open Services Gateway Initiative*)

L'*Open Services Gateway Initiative*⁴⁰ (OSGi) est une organisation indépendante dans le but de promouvoir des spécifications ouvertes pour la fourniture et la gestion de services dans des réseaux résidentiels ou autres types d'environnement restrictif : véhicules de transport, contrôle industriel... OSGi définit une plateforme Java centralisée de services avec un seul répertoire des services [OSGi, 2004]. Dans OSGi, les services sont fournis et déployés dans des unités logiques et physiques appelées *Bundles*. Du point de vue logique, un *bundle* est un fournisseur et/ou demandeur des services. Du point de vue physique, un *bundle* correspond à un fichier JAR contenant du code Java et des ressources. OSGi fournit des mécanismes pour gérer le cycle de vie d'un bundle permettant de créer, détruire, lancer, arrêter et mettre à jour ces composants dynamiquement.

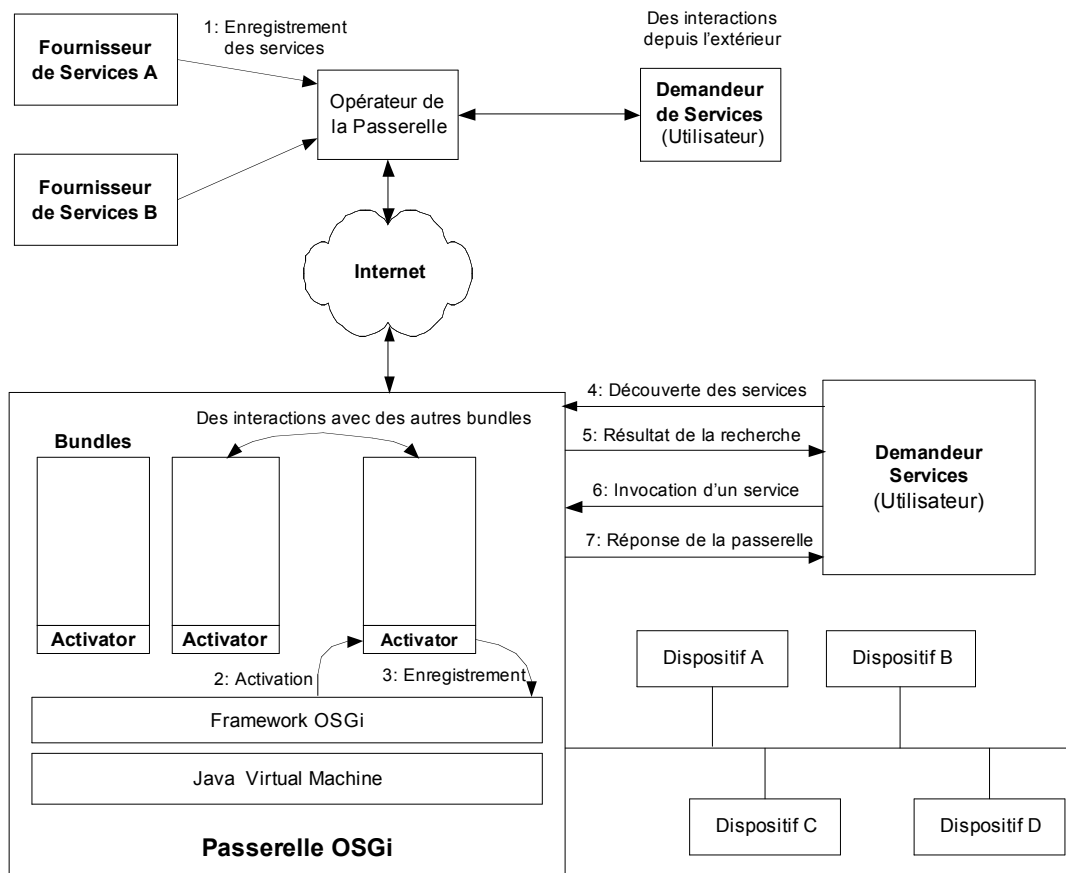


Figure 49 : Architecture OSGi.

La Figure 49 montre l'architecture générale de l'architecture OSGi permettant de réaliser les opérations suivantes :

1. Les fournisseurs des services publient les interfaces Java et les propriétés associées des services dans un répertoire. L'enregistrement retourne une référence permettant de retirer le service du répertoire.
2. La classe « activateur » du framework OSGi active un *bundle*.
3. Lorsqu'un bundle est actif, il peut enregistrer ses propres services, rechercher des services dans le registre de services fourni par la plate-forme, accéder à d'autres *bundles* et installer des *bundles* additionnels.

⁴⁰ www.osgi.org - OSGi Alliance

4. Les clients font la découverte des services par l'interrogation du répertoire en utilisant le nom de l'interface désirée et un éventuel filtre LDAP (*Lightweight Directory Access Protocol*).
5. Le répertoire réalise une sélection des fournisseurs de services sur la base de la liste de propriétés informées au moment de l'annonce du service, et ensuite, il envoie une réponse au client (demandeur de services).
6. Après la découverte de services, les clients peuvent invoquer les services sur la plateforme.
7. Ensuite, une réponse de la passerelle comportant le résultat est envoyée au client. En plus, OSGi permet aux demandeurs de service de s'enregistrer pour recevoir des événements indiquant des changements dans le service : service enregistré, service retiré, service modifié.

A ce jour, l'implémentation d'OSGi est dépendante du langage *Java*. L'architecture OSGi est bien adaptée pour communiquer à distance un ensemble des entités avec des demandeurs de services (utilisateurs) afin de contrôler un environnement donné. Dans ce sens, [Desertot et *al.*, 2005] proposent un système basé sur l'architecture OSGi pour surveiller à distance les mesures obtenues par des capteurs communicants dans un environnement donné et réagir en fonction de ces valeurs. La plateforme OSGi permet d'intégrer d'autres technologies existantes telles que les Services Jini ou UPnP [OSGi, 2004].

3.3.4 CORBA (*Common Object Request Broker Architecture*)

CORBA est un système distribué défini par l'OMG⁴¹ (*Object Management Group*) qui considère un réseau dynamique de courtiers, appelé fédération, en collaborant entre eux pour répondre à une requête réalisée par un demandeur d'un service. CORBA est basé sur un bus logiciel appelé ORB (*Object Request Broker*). L'ORB est l'élément central de l'architecture CORBA, c'est lui qui est en charge d'assurer les collaborations entre tous les objets CORBA. Il offre un environnement d'exécution aux objets masquant l'hétérogénéité liée aux langages de programmation, aux systèmes d'exploitation, aux processeurs et aux réseaux [OMG, 2004]. La Figure 50 montre le modèle OMA [Geib et *al.*, 1999] définissant l'architecture de gestion des objets CORBA. Ici, les Services CORBA représentent l'ensemble des objets serveurs disponibles pour développer des fonctionnalités de base : nommage, courtage, événements et notifications, transactions... Le service de nommage permet de localiser un objet CORBA. Le service de courtage permet de rechercher un type d'objet CORBA en fonction de ses caractéristiques en utilisant un courtier CORBA. Un courtier CORBA est un répertoire de services. Le service d'événements permet aux objets de produire des événements asynchrones à destination des objets consommateurs à travers des canaux d'événements. Aux consommateurs sont notifiés uniquement des événements qui les intéressent. Le service de transaction assure l'exécution de traitements transactionnels des objets distribués et des bases de données. Afin de faciliter la relation utilisateur – système, le modèle OMA (*Object Management Architecture*) considère des ressources ou fonctions spécialisées (*facilities*) telles que l'interface utilisateur, la gestion de l'information, l'administration, etc. Les interfaces de domaine définissent des interfaces spécialisées répondant aux besoins spécifiques d'un domaine d'application déterminé : télécommunication, industrie, ... Le protocole de transport GIOP (*General Inter-ORB Protocol*) permet d'assurer la communication entre des objets distants. Pour implémenter l'architecture CORBA sur Internet, le protocole IIOP (*Internet Inter-ORB Protocol*) est utilisé.

⁴¹ www.omg.org - *Object Management Group*

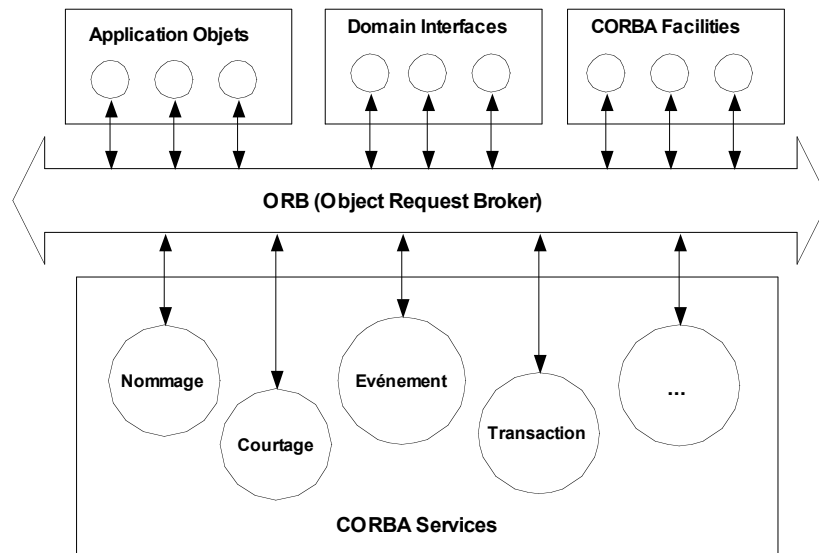


Figure 50 : Architecture CORBA et Modèle *Object Management Architecture* (OMA)

Dans CORBA, la description des services se fait dans un langage de description d'interfaces (IDL). Cette description contient une référence vers l'interface de service ainsi qu'un ensemble de propriétés obligatoires ou optionnelles caractérisant le service. Les propriétés d'un service doivent être créées et enregistrées avant que les fournisseurs puissent publier leurs services. Lors de l'annonce d'un service, publiée par un fournisseur de services, les propriétés standard définies auparavant doivent être remplies. La découverte dans CORBA permet de faire des requêtes sophistiquées basées sur les propriétés déclarées dans la description du service, des préférences permettant de changer l'ordre des réponses et des politiques destinées à limiter la propagation d'une requête dans une fédération de courtiers. Le répertoire retourne un ensemble d'offres contenant des références vers les objets fournissant le service.

[Sanz et Alonso, 2001] considèrent que l'architecture CORBA est adéquate pour développer des applications dans le cadre des systèmes de contrôle distribué. Les auteurs justifient cette affirmation en signalant que l'architecture supporte le comportement réactif et proactif des objets en temps réel, en utilisant des objets de nature hétérogène dans différentes plate-formes de travail (*hardware, software*). On peut citer des applications basées sur CORBA pour télécommander des robots (fixes et mobiles) et des caméras à distance dans un environnement distribué indépendant du langage de programmation et du système d'exploitation utilisé par les demandeurs de services [Jia et al., 2004].

3.3.5 Les Services Web

Les Services Web ou « *Web Services*⁴² », apparaissent aujourd'hui comme le nouveau paradigme des architectures logicielles et représentent une technologie permettant à des applications de dialoguer à distance par Internet indépendamment des plates-formes matérielles et des langages d'implémentation, selon une approche d'invocation de composants applicatifs à distance. Ils s'appuient sur une architecture distribuée, en considérant de multiples répertoires de services [Kadima et Monfort, 2003] afin de permettre l'interopérabilité entre différentes applications hétérogènes pouvant s'exécuter au sein de différents réseaux d'entreprises. Dans les services web, le langage WSDL (*Web Service Description Language*), basé sur XML, est utilisé pour décrire les services [W3C, 2004]. Une

⁴² www.w3.org/2002/ws - *Web Services Activity (W3C)*

description de service contient l'interface du service, les types de données employés dans les messages, le protocole de transport employé dans la communication et des informations permettant de localiser un service spécifié. Le répertoire des services s'appelle UDDI (*Universal Description Discovery and Integration*). En conséquence, UDDI supporte l'enregistrement des descriptions de service ainsi que l'enregistrement des fournisseurs des services. UDDI est un registre distribué dans lequel l'information est répliquée sur plusieurs sites. Le répertoire UDDI permet de retourner l'information sur un ou plusieurs fournisseurs de services (*find_business*) et retourne l'information sur les services fournis par les fournisseurs de services enregistrés (*find_service*). La découverte peut retourner les résultats, de façon ordonnée, en suivant plusieurs critères. En plus, UDDI supporte l'enregistrement auprès du répertoire pour recevoir des notifications concernant les changements dans le répertoire (l'ajout, le retrait et les modifications au niveau des services ou des entreprises). Le protocole de communication par défaut est SOAP (*Simple Object Acces Protocol*).

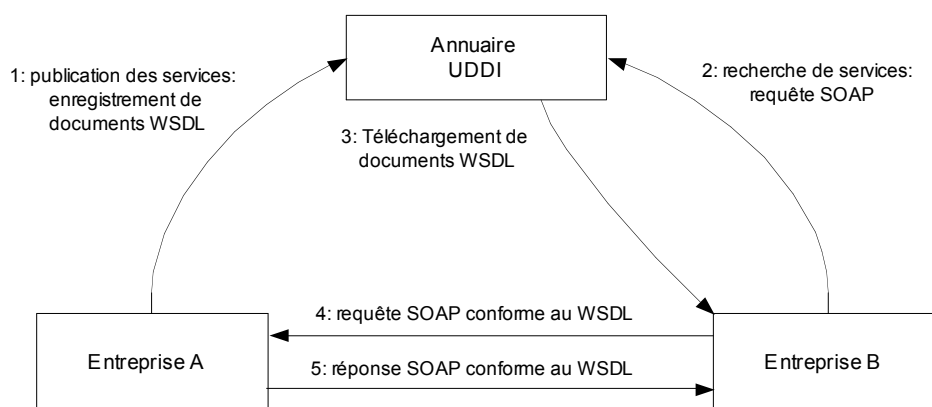


Figure 51 : Fonctionnement générale des Services Web.

La Figure 51 illustre le fonctionnement général des services Web. Dans cette illustration, on observe que :

1. L'entreprise A publie ses services dans l'annuaire UDDI.
2. L'entreprise B interroge l'annuaire afin de savoir s'il a des renseignements sur un service déterminé.
3. Si la réponse est affirmative, l'entreprise B télécharge les documents de description de services.
4. Donc, l'entreprise B peut communiquer directement avec l'entreprise A. Les invocations de services se font par requête SOAP conforme à la description donnée par l'annuaire UDDI.

Selon [Wang et *al.*, 2004] les concepts clés pour démocratiser l'usage des services Web, dans les entreprises, sont la sécurité des transactions, la composition des services et la sémantique des services pour, ainsi, offrir des services plus complexes.

Pour augmenter la fiabilité des transactions il y a actuellement de nouveaux standards proposés. Dans ce sens, les efforts les plus importants sont :

- XML Signature [W3C, 2003] pour représenter des signatures digitales afin de signer entièrement ou partiellement les documents XML ;
- XML Encryptage [W3C, 2001] pour crypter et décrypter un document XML ;
- WS-Security [OASIS, 2003] pour offrir des fonctions d'autorisation d'accès et de chiffrement des échanges ;
- SAML (*Security Assertion Markup Language*) [OASIS, 2004] pour authentifier et habiliter les transactions SOAP ;

- et finalement, XACML (*eXtensible Acces Control Markup Language*) [OASIS, 2001] pour décrire les politiques / droits d'accès à des objets XML.

Afin de décrire les processus d'une entreprise, les états de ses services Web et sa composition interne de manière standardisée, il existe différentes initiatives, par exemple BPEL4WS (*Business Process Execution Language for Web Services*), WSCI (*Web Services Choreography Interface*) et BPML (*Business Process Management Language*). BPEL4WS est basé sur XML et décrit comment les opérations peuvent être séquencées. Pour cela, le terme orchestration est employé pour modéliser le comportement des entités dans des interactions spécifiques à une entreprise. Complémentairement, le terme chorégraphie est employé pour modéliser la séquence de messages publics échangés entre les différentes parties. WSCI est employé pour décrire le comportement observable ou visible entre les services Web sans décrire les processus internes de l'entreprise. BPML est un métalangage pour décrire les processus métiers internes d'une entreprise prenant en compte les activités pour envoyer, recevoir et invoquer des services Web. Il est important de souligner qu'à ce jour, BPEL4WS est le standard de-facto utilisé par les entreprises qui fournissent des solutions Web dans la gestion de processus d'affaires [Leymann et Roller, 2005]. Il est agréé par l'OASIS (*Organization for the Advancement of Structured Information Standards*) [OASIS, 2003].

Finalement, comme ligne de travail actuel et futur, nous pouvons citer les services Web sémantiques. Ceci dans le but de décrire les fonctionnalités d'un service d'une façon complètement interprétable par des machines et améliorant, de cette façon, la qualité des fonctionnalités de découverte, d'invocation, de composition, de surveillance et d'actualisation des services Web. Pour représenter les ontologies, le W3C propose le langage OWL (*Ontology Web langage*) [W3C, 2004]. OWL est un langage d'ontologie Web utilisé pour publier et partager des données sur Internet en utilisant des ontologies, c'est-à-dire, en utilisant des terminologies spécifiques (des concepts et des propriétés) pour décrire des domaines d'application concrets.

3.4 Analyse et synthèse comparative des architectures de services ambiants présentées

Nous pouvons dire qu'en général une architecture de services ambiants permet aux dispositifs de découvrir et offrir leurs services dans un réseau ambiant et pouvant être étendus au réseau Internet. Dans le but de réaliser des interactions entre demandeur et fournisseurs de services, il est nécessaire d'identifier des critères de choix et de classification expliquant leur pertinence par rapport à notre problématique de gestion de produits communicants dans la chaîne logistique.

- Critère 1 - Architecture distribuée de services ambiants dans des répertoires centraux :

En général, nous pouvons dire qu'il y a deux types d'architectures distribuées : les architectures qui utilisent un ou plusieurs répertoires centraux et les architectures qui n'utilisent pas d'intermédiaires pour établir la communication entre les entités. Une architecture distribuée, avec un ou plusieurs répertoires de services pour assurer l'interaction entre entités, est capable de gérer le traitement des requêtes complexes effectuées par des clients. En effet, un répertoire central a une vision globale et complète de toutes les ressources disponibles dans l'environnement. Un répertoire de services centralisé stocke toute l'information des services disponibles dans un endroit physique unique. L'usage d'un répertoire centralisé a deux faiblesses principales : être un possible goulot d'étranglement pour le transport de l'information entre des entités, et être un point de panne potentiel, ou de vulnérabilité, en impliquant l'arrêt de tout le système en cas de répertoire centralisé unique. S'il existe plusieurs répertoires centraux, le système est plus robuste mais le risque de panne continue en affectant le fonctionnement du système. Ce type de configuration est adapté à des scénarios dans

lesquels il existe plusieurs alternatives de services pour un utilisateur final, et la défaillance d'une partie du système n'implique pas son arrêt total en lui permettant l'accès à un service alternatif pour développer la tâche souhaitée.

Dans une architecture distribuée sans répertoire central, les services sont répartis sur l'ensemble des entités. Chaque fournisseur de services peut répondre à des requêtes en considérant seulement leurs services. En effet, ils ne connaissent pas la totalité des services disponibles dans l'environnement. Dans ce cas-là, c'est le demandeur du service qui est chargé d'évaluer la meilleure option de service pour accomplir sa tâche. Dans une architecture complètement distribuée, les échanges d'information sont plus rapides car plus directs, le système résiste mieux aux risques de pannes car les services sont distribués dans chaque entité, et il y a un usage des capacités de calcul et de traitement de l'information (processeur, mémoire CPU, stockage, ...) mieux équilibré. En conséquence, une architecture purement distribuée est plus robuste qu'une architecture distribuée ayant un ou plusieurs répertoires centraux. De plus, ce type d'architecture, pair à pair (P2P) distribuée, doit supporter dynamiquement un accroissement du nombre d'entités intégrant un système, l'évolution et la mobilité de ces entités dans le temps. Une architecture de ce type doit être capable de gérer l'hétérogénéité des différents *hardwares* et *softwares* utilisés par les ressources. La Figure 55 montre une architecture P2P distribuée avec cinq paires communiquent entre elles ne nécessitant aucun répertoire central pour interagir. Comme inconvénient nous citons le fait qu'une architecture distribuée sans répertoire central consomme plus de bande passante qu'une architecture distribuée avec un répertoire central [Govea et Barbeau, 2000]. Pour toutes ces raisons, nous considérons qu'une architecture entièrement distribuée est la plus adéquate pour développer des interactions entre entités de type produits communicants.

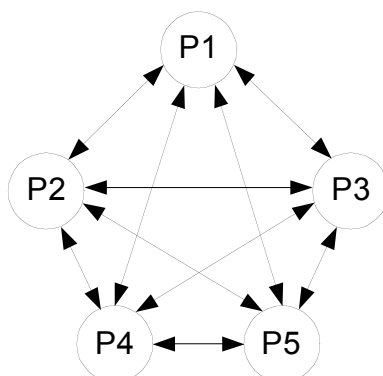


Figure 52 : Exemple d'une architecture réseaux P2P distribuée.

Par rapport aux architectures étudiées, nous pouvons dire que : Jini est une architecture distribuée avec un ou plusieurs répertoires centraux (*Service Lookup*) ; UPnP est une architecture distribuée sans répertoire central ; OSGi est une architecture centralisée avec un seul répertoire centralisé ; CORBA est une architecture distribuée avec plusieurs répertoires centraux (réseau de courtiers) ; et l'architecture de services Web est une architecture distribuée avec plusieurs répertoires centraux.

- Critère 2 - Mode mixte pour la découverte des services dans un réseau ambiant :

Comme deuxième critère, nous spécifions le type de découverte des services dans un réseau ambiant. La découverte automatique de services est l'élément clé d'une infrastructure de services. Ainsi, les dispositifs peuvent apprendre quels services sont disponibles dans un réseau ambiant. Il y a deux mécanismes de base pour faire cette tâche : la découverte passive et la découverte active de services. L'annonce de services, par un fournisseur de services sur un réseau ambiant, est associée à une

découverte passive de service. Ainsi, le demandeur de service est informé de la présence de nouveaux services dans un réseau ambiant. En revanche, dans une recherche active, le demandeur de services fait une requête lorsqu'il a besoin d'un service. Nous considérons que le mode mixte de recherche, combinant les modes de découverte passive et active, est le plus adéquat pour un environnement dynamique et spontané, tel qu'une population de produits communicants.

Par rapport aux architectures analysées nous pouvons préciser que : dans Jini et UPnP les demandeurs de services emploient un mode de recherche mixte pour découvrir les services ; dans OSGI, CORBA et les Services Web les demandeurs de services utilisent le mode de recherche active pour découvrir les services offerts par les fournisseurs de services.

- Critère 3 - Méthode de description de services :

Ce critère exige une riche description de services afin de faciliter sa future recherche et son usage. Les méthodes pour décrire les services sont associées à la façon dont ceux-ci seront détaillés. Les demandeurs de services feront des requêtes en fonction des possibilités offertes par la méthode de description employée et sa mise en place. Dans Jini, une classe décrit l'identificateur, le type, et les valeurs des attributs d'un service. Le critère pour évaluer une demande de service dans Jini est basé sur l'égalité entre le service demandé et le service enregistré, et entre ses propriétés indiquées au moment de la requête et celles contenues dans la description du service. Dans UPnP la description des services, réalisée sur des documents XML, permet de faire des recherches en considérant le type de service souhaité. Dans OSGi la méthode basée sur Java obtient les références des services vers les fournisseurs. Si une référence est activée par un demandeur de service, le protocole LDAP permet de sélectionner les fournisseurs de services selon la liste de propriétés informées au moment de sa publication. Dans CORBA, la description des services basée sur le langage IDL contient une référence vers l'interface du service ainsi qu'un ensemble de propriétés caractérisant le service. Dans ce dernier cas, les requêtes peuvent contenir des filtres logiques et mathématiques afin d'affiner la recherche d'un service. Dans les Services Web, les services sont décrits en langage WSDL (basé sur XML). Les requêtes peuvent s'effectuer selon différents critères afin d'obtenir de l'information sur les services fournis par les entreprises enregistrées auprès de l'annuaire.

- Critère 4 - Méthode d'invocation de services basée sur le transfert de données :

Comme quatrième critère de choix, nous considérons la méthode employée par les différentes architectures, afin d'invoquer et d'exécuter les services ambiants. On peut distinguer deux méthodes génériques pour cela : l'invocation de services basée sur la mobilité de code et l'invocation de services basée sur la mobilité de données (approche protocole). Dans Jini, OSGi et CORBA le code est mobile entre les entités. Ceci implique de disposer d'une plateforme d'exécution standardisée et partagée pour toutes les entités. Comme point faible de cette méthode, nous citons les problèmes de sécurité associés à la mobilité de code au niveau application. Dans UPnP et les Services Web, les entités échangent des données qui sont ensuite interprétées et utilisées par les entités serveur. Nous considérons que cette dernière approche est très bien adaptée à nos exigences parce que cette approche permet de faire communiquer les entités informatiques sur la base d'échanges de fichiers de données standardisés de petite taille. Ainsi, on supprime les problèmes de vulnérabilité (code malveillant, virus, ...) associés à la mobilité du code.

- Critère 5 - Notifications d'événements entre dispositifs :

Les messages de notifications entre entités permettent de transmettre aux clients les changements d'état des services. Dans un environnement dynamique, cette exigence est fondamentale. Il faut

connaître à tout moment quels services sont effectivement disponibles dans un réseau ambiant. Les architectures étudiées incorporent des méthodes pour informer les demandeurs de services si un changement d'état dans un service se produit.

- Critère 6 - Environnement de développement :

L'environnement logiciel est un élément important au moment d'implémenter la notion de service ambiant. Nous considérons qu'une architecture de service ambiant doit être indépendante du langage de programmation et du système d'exploitation utilisé. Toutes les architectures étudiées sont indépendantes du système d'exploitation. Jini est dépendant du langage Java (exécution sur une machine Java JVM). CORBA est non dépendant du langage de programmation et dépendant du protocole de transport GIOP ou IIOP. UPnP et les services web sont non dépendants du système d'exploitation et du langage de programmation. Ces deux architectures, de nature différente, reposent sur les protocoles standards Internet TCP/IP, HTTP et SOAP.

Dans le Tableau 4 nous présentons les trois niveaux de satisfaction, que nous avons définis, afin d'évaluer la pertinence de chaque architecture par rapport à chaque critère de choix dans notre contexte d'étude.

Symbole	Signification
-1	Critère non satisfait par l'architecture
0	Critère satisfait partiellement par l'architecture
+1	Critère satisfait entièrement par l'architecture

Tableau 4 : Critères d'évaluation des architectures de services ambiants.

Dans le Tableau 5 nous présentons la synthèse comparative des architectures de services analysées. En fonction des critères de choix définis et de l'analyse comparative réalisée, l'architecture qui nous semble la mieux adaptée pour atteindre nos objectifs est l'architecture UPnP. Les six critères de choix définis sont satisfaits entièrement par l'architecture de services ambiants UPnP. La standardisation des protocoles utilisés par UPnP tels que SSDP, SOAP et GENA, alliés à l'usage extensif de XML, comme langage de description, est un atout majeur pour la sélection de cette architecture. Il est majeur d'indiquer que l'architecture UPnP repose sur un principe de Zéro configuration au moment des interactions entre des entités, en permettant la découverte automatique des services ambiants sans besoin de configuration administrative.

	Jini	UPnP	OSGi	CORBA	WebServices
Type d'architecture	Système distribué avec des répertoires centraux.	Système distribué sans répertoires centraux.	Système centralisé avec un seul répertoire centralisé.	Système distribué avec des répertoires centraux.	Système distribué avec des répertoires centraux.
	0	+1	-1	0	0
Mode de découverte de Services	Mixte	Mixte	Active	Active	Active sur une URL d'un service.
	+1	+1	0	0	0
Méthode de description de services	Interface Java et un nombre variable de propriétés.	Documents XML	Interface ou classe Java et un nombre variable de propriétés.	Langage de description d'interfaces IDL.	Langage WSDL basé sur XML
	+1	+1	+1	+1	+1
Méthode d'invocation de services	Mobilité du code (Java RMI)	Mobilité des données (SOAP)	Mobilité du code (Java RMI)	Mobilité du code (IIOP)	Mobilité des données (SOAP)
	-1	+1	-1	-1	+1
Notifications des événements	Oui (pour l'arrivée, le départ et le changement d'un service)	Oui (GENA)	Oui (service enregistré, service retiré, service modifié)	Oui (à implémenter sur l'courtier CORBA)	Oui (ajout, retrait et modification des services ou des entreprises)
	+1	+1	+1	+1	+1
Environnement de développement	Plateforme Java sur JVM. Non dépendant du OS, dépendant du langage de programmation Java et non dépendant du protocole de transport réseau.	Non dépendant du OS et du langage de programmation. Dépendant du protocole de transport réseau (TCP/IP, HTTP, SOAP) Plateforme dépendant du SDK (Software utilisé).	Plateformes Java sur JVM. Non dépendant du OS, dépendant du langage de programmation Java et non dépendant du protocole de transport réseau.	Non dépendant du OS et du langage de programmation. Dépendant du protocole de transport GIOP ou IIOP.	Non dépendant du OS et du langage de programmation. Dépendant du protocole de transport réseau (TCP/IP, HTTP, SOAP)
	-1	+1	-1	+1	+1
Evaluation	1	6	-1	2	4

Tableau 5 : Synthèse comparative des architectures de services étudiées.

3.5 Analyse de l'architecture *Universal Plug and Play* (UPnP)

L'architecture UPnP inclut deux types d'entités logiques : dispositifs et points de contrôle [UPnP Forum, 2003]. Le concept dispositif est utilisé pour définir une entité qui offre des services aux points de contrôle dans un réseau ambiant. Les dispositifs peuvent contenir un ou plusieurs dispositifs embarqués. Chaque dispositif peut contenir un ou plusieurs services. Les services sont des entités logiques qui représentent des fonctionnalités disponibles dans un réseau ambiant. Chaque service est composé d'un ensemble d'actions permettant de modifier l'état d'un dispositif. Les points de contrôle sont des entités logiques que peuvent découvrir d'autres dispositifs, et après, invoquer les services offerts. Un point de contrôle peut contenir des dispositifs. Dans ce cas, le point de contrôle présente un rôle dual, étant à la fois, dispositif et point de contrôle.

3.5.1 Entité Dispositif UPnP

Un dispositif UPnP est un conteneur logique de services et de dispositifs embarqués qui sont contrôlés par des points de contrôle. Dans le cas où un dispositif est composé d'un ensemble de dispositifs embarqués, on utilise le terme « dispositif racine » pour le désigner. Au niveau de son adressage logique IP, tous les sous-dispositifs auront la même adresse IP, celle assignée au dispositif racine. Ainsi, par exemple, un produit communicant peut être représenté comme un dispositif racine offrant ses services intrinsèques et les services associés à ses sous-dispositifs tels que des capteurs intégrés de température, humidité..., des actionneurs, etc. Toutes les informations qui concernent la description d'un dispositif sont capturées dans un fichier XML. Ce fichier XML liste aussi les services associés au dispositif et les propriétés liées à ce dispositif. Le fichier XML contient la liste de dispositifs embarqués et une URL pour accéder à la page HTML de présentation du dispositif.

3.5.1.1 Services, actions, variables d'état et événements

Le service est une entité logique permettant l'accès à des fonctionnalités spécifiques sur un réseau UPnP. Les services sont contrôlés par des points de contrôle. Un service exprime des actions et caractérise son état par des variables d'états. Un service dans un dispositif UPnP est constitué d'une table d'état, d'un serveur de contrôle, et d'un serveur d'événements. La table d'état modélise l'état du service, à travers des variables d'états, et les met à jour lorsque l'état change. La partie serveur de contrôle reçoit des demandes d'actions, les exécute, met à jour la table d'état et retourne les réponses. Le serveur d'événements publie les événements aux souscripteurs intéressés lorsque l'état d'un service change. La liste des services offerts est obtenue à partir de la description du dispositif dans un fichier XML. La description détaillée des services disponibles est obtenue à partir d'un ou plusieurs fichiers XML.

Une action est une commande présentée par un service qui peut être appelée à distance par des points de contrôle. Un service a, typiquement, plusieurs actions. Chaque action a un ensemble de paramètres d'entrée ou de sortie et, optionnellement, une valeur de retour. Une variable d'état, maintenue par un service, est caractérisée par un nom, type, valeur par défaut, intervalle de valeurs, et peut déclencher des événements quand sa valeur change. Des variables d'état peuvent être employées pour modéliser l'état d'un dispositif physique. Un service envoie une notification d'événement aux points de contrôle quand une variable d'état change. Des variables d'état sont marquées comme événementielles ou non dans le document de description du service.

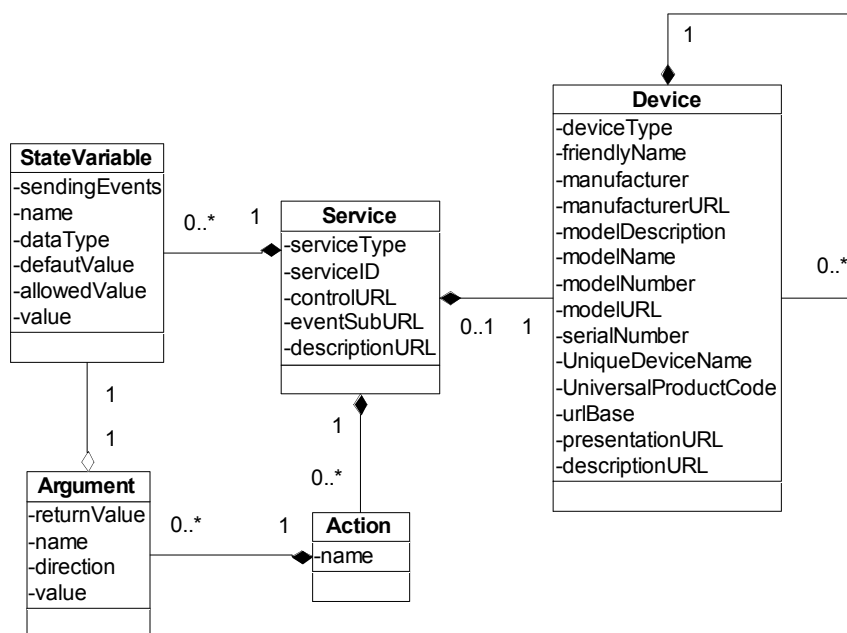


Figure 53 : Diagramme de classe décrivant un dispositif UPnP.

Le diagramme de classe présenté dans la Figure 53 montre la relation entre un dispositif et ses services [Jeronimo et Weast., 2003]. La figure illustre un dispositif UPnP associé à un ensemble de services. Ces services sont décrits par des variables d'état. Un service est associé à un ensemble d'actions. Ces actions permettent d'accéder aux valeurs des variables d'état en utilisant de paramètres d'appel.

3.5.2 Entité Point de contrôle UPnP

Un point de contrôle est une entité logique capable de découvrir et de contrôler d'autres dispositifs [UPnP Forum, 2003]. Après la phase de découverte, un point de contrôle peut : récupérer la description d'un dispositif, récupérer la liste des services associés à un dispositif, récupérer la description d'un service parmi ceux qui l'intéressent, invoquer des actions pour contrôler / agir sur le dispositif, souscrire aux événements d'un service. Lors d'un changement de l'état d'une variable d'un service, le serveur d'événements notifiera l'événement au point de contrôle.

3.5.3 Entité mixte : Dispositif UPnP et Point de Contrôle UPnP

Un point de contrôle peut contenir des dispositifs et vice-versa. Dans ce cas, le point de contrôle et le dispositif présentent un rôle dual. Ils auront un rôle de demandeurs et fournisseurs de services dans un réseau de services ambiants UPnP.

3.5.4 Modèle de la pile des protocoles réseau dans UPnP

Les protocoles TCP/IP et HTTP fournissent la connexion réseau ambiant de base pour l'architecture de dispositifs UPnP. Chaque dispositif fournit un ensemble d'URLs HTTP pour que les points de contrôle puissent rechercher le document de description du dispositif, contrôler le dispositif, souscrire aux événements, et obtenir une page de présentation des dispositifs. L'architecture UPnP permet qu'un serveur communique avec plusieurs destinataires simultanément. À cet égard, l'architecture UPnP

emploie un message HTTP qui est envoyé sur une connexion réseau *unicast* ou *multicast*. La Figure 54 montre la pile de protocoles employés par UPnP. On observe que les communications ambiantes peuvent être réalisées en utilisant des réseaux de communication Ethernet ou Wifi sur les protocoles IP. La découverte des dispositifs est réalisée sur le protocole UDP (*User Datagram Protocol*). La description des dispositifs et leurs services, et la gestion des événements, sont réalisés en mode connecté par le protocole TCP (*Transmission Control Protocol*) tandis que les mécanismes de découverte font appel à UDP. Chaque fonctionnalité au sein d'UPnP a besoin de protocoles spécifiques tels que SSDP, SOAP et GENA. Ils seront expliqués dans les sections suivantes de ce manuscrit.

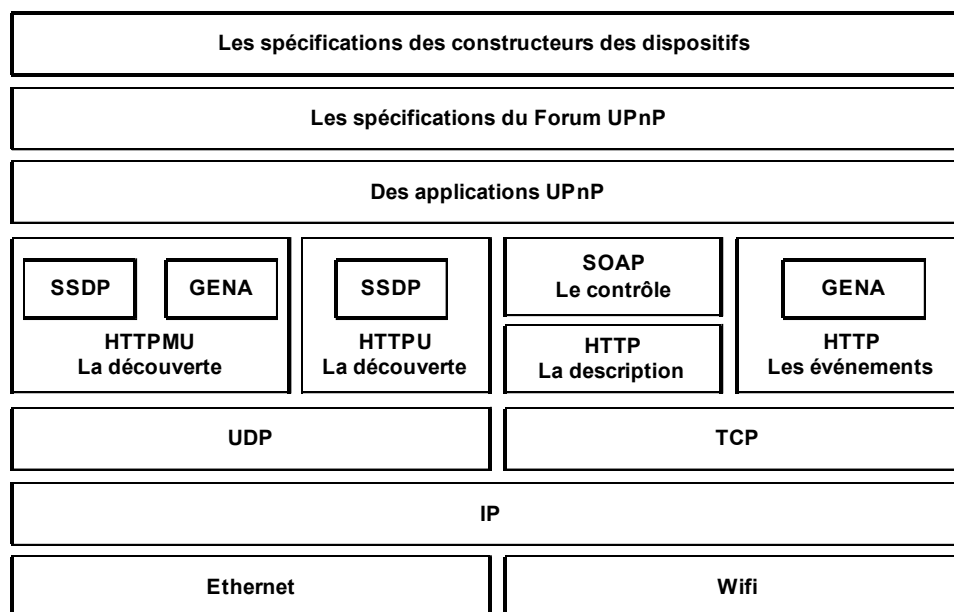


Figure 54 : Pile de protocoles réseau dans UPnP.

3.5.4.1 HTTP *multicast* et HTTP *unicast*

L'architecture UPnP est basée sur le protocole de communication informatique client - serveur HTTP (*HyperText Transfer Protocol*). Egalement, UPnP utilise IP *unicast* ou *multicast* comme moyen de transport pour envoyer des messages, en temps réel, vers un destinataire ou plusieurs destinataires respectivement. Un point de contrôle utilise un mécanisme de découverte de service basé sur IP *multicast* afin de rechercher les dispositifs présents sur le réseau ambiant. Un dispositif annonce sa présence sur le réseau ambiant en utilisant des messages sur IP *multicast*. Les services envoient des notifications d'événements sur IP *multicast* vers les points de contrôle qui ont souscrit pour les recevoir.

Dans UPnP, la communication *unicast* et *multicast* est faite en utilisant le protocole HTTP sur UDP (*User Datagram Protocol*) au lieu de TCP (*Transmission Control Protocol*). La communication sur UDP est plus rapide que la communication sur TCP. Par exemple, avec HTTPU sur UDP *unicast*, un dispositif peut envoyer un message HTTP vers un point de contrôle plus rapidement qu'en utilisant une connexion TCP. HTTPMU sur UDP *multicast* permet d'envoyer des messages HTTP à plusieurs destinataires (groupe) simultanément. Chaque serveur, appartenant au groupe logique de communication, partage une adresse commune de *multicast* avec le but de recevoir les données envoyées à cette adresse. Un mécanisme permet aux serveurs de joindre ou non ce groupe. Les dispositifs et points de contrôle UPnP utilisent l'adresse *multicast* 239.255.255.250 et le port 1900. En

utilisant cette adresse et port, il est possible qu'un dispositif quelconque puisse envoyer des données simultanément vers les autres dispositifs et points de contrôle connectés au réseau ambiant.

3.5.4.2 XML (*Extensible Markup Language*)

XML est un méta langage qui sert de base pour créer des langages balisés afin de structurer des données [W3C, 2004]. Il utilise des balises et des attributs pour décrire les données d'une manière hiérarchique. XML est employé par l'architecture UPnP pour décrire les dispositifs et les services qu'ils fournissent, pour exprimer les contenus des messages entre dispositifs, et pour spécifier le format des messages d'événements envoyés par les services aux points de contrôle.

3.5.4.3 SSDP (*Simple Service Discovery Protocol*)

SSDP est le protocole de découverte de services utilisé par l'architecture UpnP, qui ne nécessite ni configuration, ni gestion administrative, ni intervention humaine pour son fonctionnement. Pour identifier un service, SSDP utilise deux concepts : le *service type* et l'*Unique Service Name* (USN). Le *service type* est un identifiant URI (*Uniform Resource Identifier*) permettant de connaître la fonction d'une ressource particulière.

Un URI est une chaîne de caractères permettant d'identifier une ressource abstraite ou physique accessible sur Internet. La syntaxe des URIs a été définie par Tim Berners-Lee et décrite en détail en août 1998 dans la RFC 2396 (*Request For Comment*) de l'IETF (*Internet Engineering Task Force*).

Exemple d'URI : <http://193.50.39.120:8080/objet/description.xml>.

Les URLs (*Uniform Resource Location*) identifient une ressource par le moyen de leur chemin d'accès et précisent donc leur situation sur le réseau ("*location*"). Comme les URL ont été le premier type d'URI répandu, le sigle URL a parfois été utilisé en lieu et place du concept d'URI.

L'USN est un URI permettant d'associer un nom unique aux services pour les différencier des services du même type. Un USN typiquement contient un UUID (*Universally Unique Identifier*) utilisé pour identifier uniquement un objet.

Il y a deux types de requêtes SSDP. La première, est une requête pour découvrir les ressources dans le réseau et la deuxième est une requête pour annoncer la présence d'une ressource dans le réseau ambiant. SSDP utilise HTTP sur UDP *multicast* (HTTPMU) pour envoyer des messages à tous les points SSDP sur le réseau ambiant. Les points de contrôle, représentant les clients SSDP, effectuent des requêtes de recherche afin de découvrir les dispositifs et leurs services à l'adresse 239.255.255.250 :1900, laquelle est réservée pour la communication *multicast*. Un dispositif, ou ressource SSDP, écoute le canal *multicast* pour s'informer sur les requêtes de découverte. Si un dispositif, reçoit une requête de découverte HTTPMU *multicast* qui est adapté aux services offerts, il répond en envoyant une réponse directe au client SSDP (point de contrôle) qui a initié la recherche, en utilisant HTTPU *unicast*. Un point de contrôle peut reformuler sa recherche de manière plus précise dans le but de trouver des dispositifs ou des services particuliers en utilisant des mécanismes de recherche adaptés pour trouver tous les dispositifs, les dispositifs racines seulement et un type de dispositif, ou bien, un service déterminé. Il y a deux types de notifications : l'arrivée d'un dispositif et le départ d'un dispositif. SSDP utilise des méthodes de notification GENA sur HTTPMU. Lorsqu'un dispositif est inséré dans le réseau, il envoie en mode *multicast* (HTTPMU) des messages annonçant les services

qu'il supporte. Lorsqu'un dispositif est enlevé du réseau, les points de contrôle sont informés au moyen d'un message de notification GENA sur HTTPMU.

3.5.4.4 SOAP (*Simple Object Access Protocol*)

SOAP [W3C, 2003] est un protocole qui rassemble XML et HTTP afin de transmettre des messages sur HTTP et fournir un mécanisme permettant de faire des appels de procédures sur un ordinateur distant, selon une méthode nommée RPC (*Remote Procedure Call*). XML est employé pour exprimer les contenus des messages SOAP, alors que le HTTP est employé pour envoyer des messages à leur destination. Le corps d'un message SOAP contient une méthode d'appel à distance avec ses arguments associés, une méthode de réponse et l'information d'erreur pour des appels échoués. Chaque requête envoyée par un point de contrôle est détaillée à l'intérieur d'un message SOAP, y compris le nom de l'action à invoquer et les arguments qui sont associés à l'action. Ceci fait, un service envoie au point de contrôle un message contenant la valeur de réponse retournée par l'action et les arguments. En cas d'erreur, le service retourne un message d'erreur au point de contrôle en indiquant le code d'erreur.

3.5.4.5 GENA (*Generic Event Notification Architecture*)

Les composants d'un système réparti communiquent généralement en utilisant deux mécanismes différents : le RPC et la notification d'événements. En RPC, les objets attendent passivement pour fournir des services aux clients. Cependant, avec la notification d'événements, les changements d'état du système sont modélisés comme des événements, permettant aux objets, informés de ces changements, de s'adapter dynamiquement afin d'obtenir un comportement global plus efficient. GENA est un protocole pour la gestion des publications et souscriptions aux notifications d'événements en utilisant HTTP. Le protocole GENA utilise HTTPMU sur UDP multicast pour notifier la présence d'un dispositif avec le protocole SSDP. GENA utilise HTTP sur TCP/IP pour signaler les changements d'état d'un service lors de la gestion d'événements UPnP. Les messages de notification contiennent le nom d'une ou plusieurs variables et la valeur actuelle de ces variables, formatées en langage XML. Un point de contrôle, intéressé par la réception de notifications d'événements, doit souscrire à une source d'événements en envoyant une requête qui inclut le service intéressé, la localisation du dispositif (qui envoie les événements) et une durée de souscription. En utilisant GENA, une souscription peut-être renouvelée périodiquement, ou bien être annulée.

3.5.5 Les fonctionnalités et mécanismes d'UPnP

L'adressage est la fonctionnalité initiale pour tous les dispositifs UPnP. Chaque dispositif doit acquérir une adresse IP avant qu'il ne puisse découvrir les autres dispositifs ou offrir ses services sur un réseau ambiant UPnP. Une fois qu'un dispositif UPnP acquiert une adresse IP, il est prêt pour fournir ses services aux points de contrôle. La Figure 55 présente le modèle interne d'un Dispositif Device UPnP.

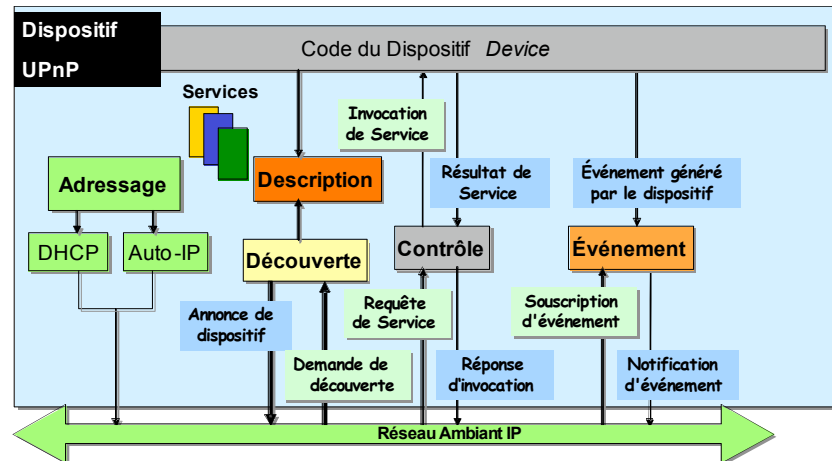


Figure 55: Modèle d'un Dispositif Device UPnP.

La découverte, la fonctionnalité suivante dans l'opération des dispositifs UPnP, permet aux points de contrôle de rechercher les dispositifs et leurs services sur le réseau ambiant et de trouver ceux qui peuvent satisfaire ses critères de recherche. Après qu'un point de contrôle UPnP a découvert un dispositif, il a seulement l'information contenue dans le message de découverte : le type de dispositif, son identificateur universel unique (UUDI), et une URL pour accéder à son document de description XML. Pour découvrir plus d'information sur un dispositif, y compris les services et les actions qu'il supporte, le point de contrôle recherche les documents de description XML du dispositif. Donc, après que le dispositif UPnP ait acquis une adresse IP et ait annoncé sa présence sur le réseau ambiant, les points de contrôle peuvent alors découvrir les dispositifs et invoquer n'importe quelle action fournie par les services d'un dispositif particulier. Dans la terminologie UPnP, ce processus d'invocation s'appelle contrôle. Après que le point de contrôle ait découvert un dispositif et ait recherché la description d'un dispositif et ses services, le point de contrôle peut souhaiter réagir aux changements d'état du dispositif. La gestion d'événements d'UPnP permet aux points de contrôle de s'inscrire auprès des serveurs d'événements pour recevoir des messages lorsque l'état d'un dispositif change. Les interactions entre les points de contrôle et les dispositifs peuvent être entièrement automatiques, n'exigeant aucune intervention humaine. Cependant, il est également possible de contrôler manuellement les dispositifs UPnP en utilisant une page web de présentation fournie par le dispositif. La Figure 56 présente le modèle interne d'un Point de Contrôle UPnP.

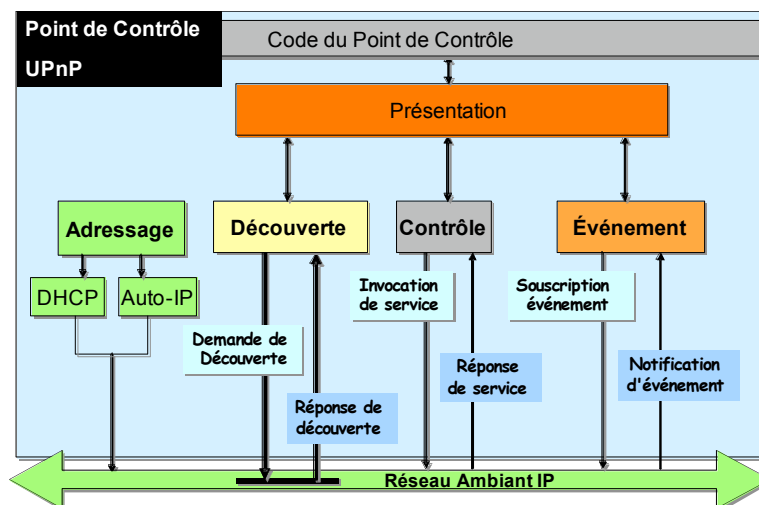


Figure 56 : Modèle d'un Point de contrôle UPnP.

3.5.5.1 L'adressage IP

L'objectif de la fonction adressage est d'assigner une adresse IP à chaque dispositif ou point de contrôle qui joint dynamiquement le réseau ambiant, soit par *Auto IP*, soit par DHCP (*Dynamic Host Configuration Protocol*). Chaque dispositif doit être un client DHCP, ou alors, il devient un serveur DHCP lorsqu'il est le premier à être connecté sur le réseau. Si un serveur DHCP est disponible, le dispositif qui intègre le réseau UPnP doit utiliser l'adresse IP qui lui a été assigné par le serveur DHCP. S'il n'y a pas de serveur DHCP sur le réseau, le dispositif doit utiliser l'Auto IP afin d'obtenir une adresse IP automatiquement. Auto IP définit la manière dont un dispositif choisit une adresse IP à partir d'une suite d'adresses privées et réservées pour cet effet.

3.5.5.2 L'annonce et la découverte d'un dispositif et ses services

Dans l'étape de découverte des dispositifs ou *devices*, UPnP utilise le protocole SSDP qui opère conjointement avec HTTPU et HTTPMU. Lorsqu'un nouveau dispositif est connecté au réseau, il annonce sa présence, ses dispositifs embarqués (si c'est le cas) et ses services, par *multicast*, aux points de contrôle, au moyen de messages de description du dispositif. Ces messages de description contiennent un URI (*Universal Resource Identifier*), qui identifie de manière unique le dispositif, et l'URL d'un fichier XML pour trouver la description détaillée du dispositif. Egalement, quand un point de contrôle est ajouté sur le réseau, le protocole de découverte lui permet de rechercher, en *multicast*, les dispositifs et les services qui les intéressent. De cette façon, tous les dispositifs écoutent et répondent si leurs services ou leurs dispositifs embarqués satisfont le critère émis dans le message de recherche. La Figure 57 représente les deux mécanismes de découverte. La partie gauche de la figure, illustre l'annonce d'un dispositif sur le réseau ambiant en utilisant le protocole SSDP sur HTTPMU (*multicast*). La partie droite de la figure illustre la recherche des dispositifs et leurs services en utilisant le protocole SSDP sur HTTPMU (*multicast*). Dans ce dernier cas, la réponse du dispositif est informée par HTTPU (*unicast*).

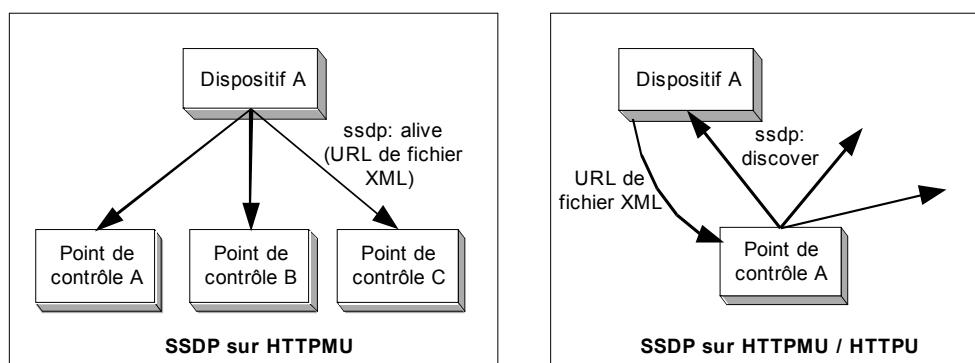


Figure 57 : L'annonce d'un dispositif et la recherche des dispositifs dans un réseau ambiant UPnP.

3.5.5.3 La description d'un dispositif et ses services

Un point de contrôle, qui a découvert un dispositif, peut obtenir plus d'information en récupérant le fichier XML de description du dispositif, grâce à l'URL informée dans le message échangé au moment de la découverte. Ce fichier contient la description du dispositif, la liste de ses dispositifs embarqués et la liste des services offerts par le dispositif. Le document de description d'un dispositif inclut des informations basiques sur le dispositif (telles que le type de dispositif, le nom du constructeur du

dispositif, l'identificateur du dispositif, la liste des dispositifs embarqués, ...), ainsi que la liste des services disponibles : le type de service, le nom du service, l'URL pour la description détaillée du service, l'URL pour le contrôle, et l'URL pour la notification d'événements et l'URL pour la présentation.

```

<?xml version="1.0"?>
<root xmlns="um:schemas-upnp-org:device-1-0">
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <URLBase>base URL for all relative URLs</URLBase>
  <device>
    <deviceType> um:schemas-upnp-org:device:DeviceType:1 </deviceType>
    <friendlyName> short user-friendly title </friendlyName>
    <manufacturer> manufacturer name </manufacturer>
    <manufacturerURL> URL to manufacturer site </manufacturerURL>
    <modelDescription> long user-friendly title </modelDescription>
    <modelName> model name </modelName>
    <modelName> model number </modelName>
    <modelURL> URL to model site </modelURL>
    <serialNumber> manufacturer's serial number </serialNumber>
    <UDN> uuid:UUID </UDN>
    <UPC> Universal Product Code </UPC>
    <iconList>
      <icon>
        <mimeType>image/format</mimeType>
        <width>horizontal pixels</width>
        <height>vertical pixels</height>
        <depth>color depth</depth>
        <url>URL to icon</url>
      </icon>
      XML to declare other icons, if any, go here
    </iconList>
    <serviceList>
      <service>
        <serviceType>um:schemas-upnp-org:service:Service1:1</serviceType>
        <serviceId>um:upnp-org:serviceId:ServiceId1</serviceId>
        <SCPDURL>URL to service description</SCPDURL>
        <controlURL>URL for control</controlURL>
        <eventSubURL>URL for eventing</eventSubURL>
      </service>
      Declarations for other services
    </serviceList>
    <presentationURL>URL for presentation</presentationURL>
  </device>
</root>

```

Des informations basiques sur le dispositif

Liste des services du dispositif

Figure 58 : Fichier XML de description d'un dispositif.

La Figure 58 illustre le document de description d'un dispositif en indiquant dans la partie supérieure, les informations basiques du dispositif et, dans la partie inférieure, la liste des services supportés par le dispositif.

La description d'un service est un document XML qui contient la liste des actions qui sont supportées par le service ainsi que les arguments pour chaque action. La description d'un service inclut aussi la liste des variables associées à chaque service du dispositif. Ces variables modélisent l'état d'un service et sont décrites selon leur type, valeur par défaut, leur intervalle de valeurs et leur caractère événementiel. La montre Figure 59 la structure d'un fichier XML pour décrire les services d'un dispositif. Les deux parties principales de ce document sont indiquées dans la Figure 59: la liste des actions supportées par le dispositif et la liste des variables qui décrivent les services du dispositif.

```

<?xml version="1.0"?>
<scpd xmlns="urn:schemas-upnp-org:service-1-0">
  <specVersion <!-- UPnP version 1.0 -->
    <major>1</major>
    <minor>0</minor>
  </specVersion>

  <actionList>
    <action>
      <name>ActionName2</name>
      <argumentList>
        <argument>
          <name> FormalParameterActionName2</name>
          <relatedStateVariable>StateVariableName2</relatedStateVariable>
          <direction>in</direction>
        </argument>
      </argumentList>
    </action>
    <Declarations for other actions>
  </actionList>

  <serviceStateTable>
    <stateVariable sendEvents="yes">
      <name>VariableName2</name>
      <dataType>variable dataType</dataType>
      <defaultValue>de fàult value</defaultValue>
    </stateVariable>
    <Declarations for other state variables>
  </serviceStateTable>
</scpd>

```

⇒ Liste d'actions

⇒ Liste de variables

Figure 59 : Fichier XML de description des services d'un dispositif.

3.5.5.4 Le contrôle d'un dispositif

Grâce à l'utilisation du protocole SOAP, les points de contrôles sont capables d'exécuter des actions à distance sur les dispositifs. La liste des actions est obtenue à partir du fichier XML qui décrit chaque service. Un point de contrôle peut envoyer un message de contrôle à l'URL de contrôle fournie par le dispositif au travers de sa description. Les messages de contrôle sont exprimés en XML. Lorsque l'action s'est terminée, le service retourne le résultat de l'action, ou une erreur en cas contraire. La Figure 60 illustre l'invocation d'une action sur un dispositif en utilisant le protocole SOAP. Dans la figure le Point de contrôle A exécute l'action (A1,1) associée au service (A,1) du Dispositif A. Ensuite, le dispositif lui envoie le résultat.

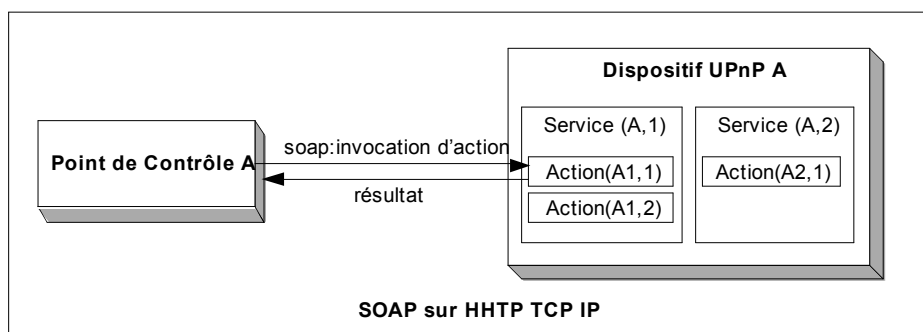


Figure 60 : L'invocation d'une action sur le dispositif.

3.5.5.5 La gestion des événements

Un point de contrôle peut s'abonner auprès des services afin d'être informé lors des changements d'état d'une variable d'état. Le service diffuse ses mises à jour en envoyant des messages d'événements aux points de contrôle souscrits. Chaque message contient les noms d'une ou plusieurs variables d'état, ainsi que leurs valeurs actuelles. Pour souscrire à un événement, un point de contrôle envoie un message de souscription à l'URL du serveur d'événements du service, qui se trouve dans le document de description du dispositif. Si la souscription est acceptée, le dispositif répond sur la durée de la souscription. Pour laisser la souscription active, le point de contrôle doit renouveler sa souscription avant que celle-ci n'expire. Le point de contrôle peut annuler la souscription à un événement. Pour gérer le *multipoint*, un message d'événement, provenant du dispositif, est envoyé à tous les points de contrôle intéressés par cet événement. Pour la gestion d'événements, le protocole utilisé est GENA. La Figure 61 illustre le mécanisme de souscription aux événements d'un service déterminé en utilisant le protocole GENA sur HTTP TCP IP. Ici, le Point de contrôle A souscrit aux événements associés avec les changements d'une variable contenue dans une table d'état. Lorsqu'un changement dans une variable concernée se produit, le point de contrôle est informé.

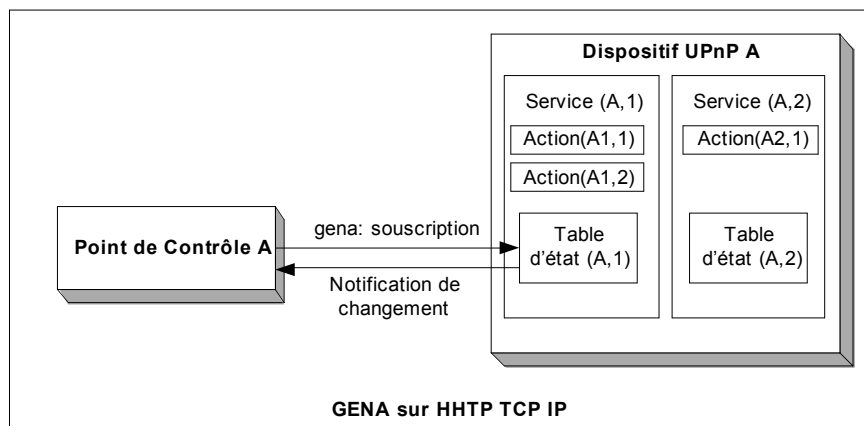


Figure 61 : La souscription aux événements.

3.5.5.6 La présentation d'un dispositif

Un point de contrôle peut charger une page HTML, ou interface d'utilisation, dont l'URL est contenue dans la description du dispositif, afin de voir l'état d'un dispositif, et ainsi, contrôler le dispositif à distance. Le point de contrôle utilise une requête GET HTTP vers l'URL de présentation et le dispositif retourne sa page de présentation.

3.5.6 Interactions entre un dispositif et un point de contrôle

Les interactions entre un dispositif et un point de contrôle peuvent être de deux types : des interactions initiées par un point de contrôle ou des interactions initiées par un dispositif.

Dans le premier cas, le point de contrôle initie une recherche d'un dispositif sur le réseau UPnP. Ce cas est illustré dans le diagramme de séquence UML présenté dans la Figure 62.

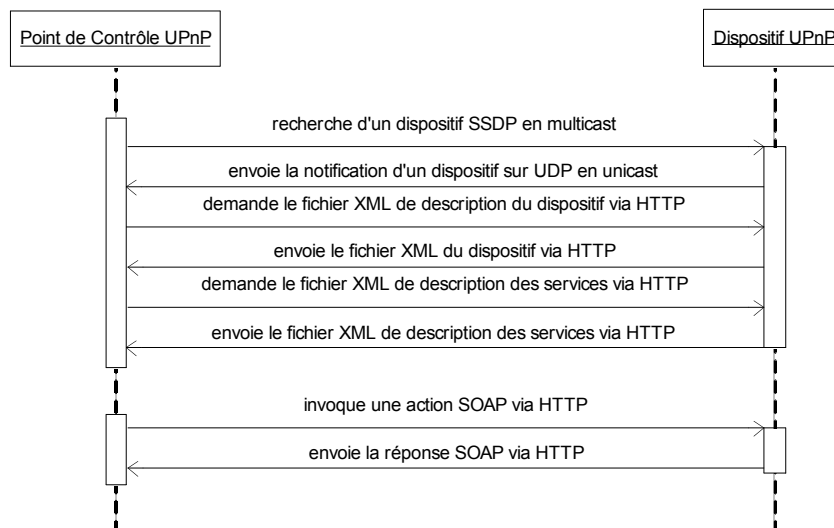


Figure 62 : Diagramme de séquence décrivant la recherche d'un dispositif et l'invocation d'une action.

On observe que le point de contrôle initie une recherche d'un dispositif en utilisant le protocole SSDP au-dessus de HTTPMU. Ensuite le dispositif concerné envoie un message de notification sur UDP, en *unicast*, en indiquant l'URL pour accéder au document XML qui décrit le dispositif. Ensuite, le point de contrôle demande le document de description XML du dispositif via HTTP. Le serveur web contenu dans le dispositif répond à la demande et renvoie au point de contrôle le document de description XML. Grâce à ce document, le point de contrôle obtient l'URL pour envoyer les demandes de contrôle au dispositif. Après ceci, le point de contrôle peut demander au dispositif d'effectuer des actions spécifiques. Après que le dispositif a exécuté l'action demandée, il envoie au point de contrôle un message de réponse SOAP au-dessus de HTTP.

Dans le deuxième cas, un dispositif initie des interactions avec un point de contrôle en annonçant sa présence sur un réseau UPnP. Cette situation est illustrée dans la Figure 63. On observe que lorsqu'un nouveau dispositif s'intègre au réseau ambiant UPnP, il annonce sa présence en informant l'URL de sa description (message SSDP) sur HTTPMU.

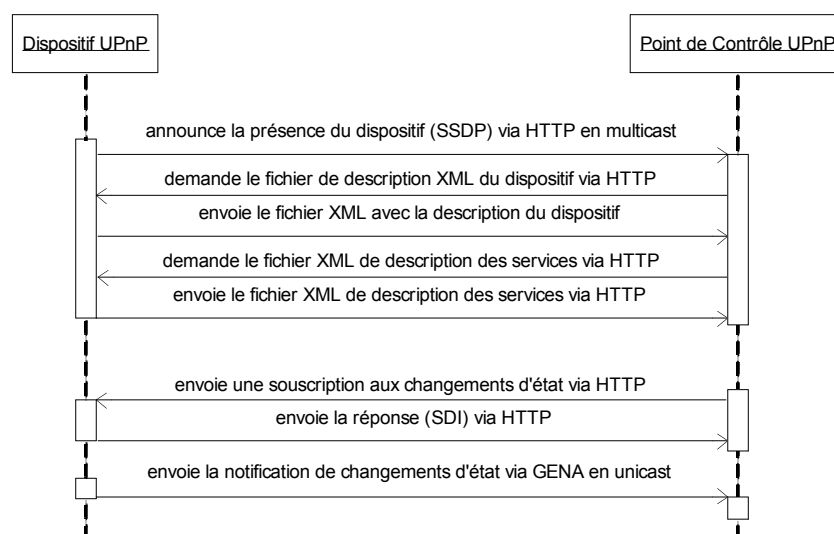


Figure 63 : Diagramme de séquence d'annonce d'un dispositif sur le réseau et souscription aux événements.

Egalement, dans la figure, on observe que le point de contrôle demande au dispositif les fichiers de description XML décrivant le dispositif et ses services. Ensuite, le point de contrôle envoie un message de souscription au dispositif (serveur d'événements) via HTTP afin de s'informer sur les changements d'état des services du dispositif. Le dispositif reconnaît la demande d'abonnement et renvoie au point de contrôle un identifiant unique d'abonnement (*Subscription Identifier* : SID). Lorsqu'un changement se produit dans l'état interne du dispositif, ce dernier envoie aux points de contrôle abonnés, un message de notification (protocole GENA), en *unicast*, au-dessus du HTTP.

3.6 Modèle conceptuel d'un objet communicant

Jusqu'ici, nous avons étudié l'état de l'art des objets communicants, les fonctionnalités génériques d'une architecture de services ambiants et les architectures de services ambiants existants les plus significatives à ce jour.

Dans cette partie du chapitre, notre intérêt est d'unifier les concepts traités antérieurement avec le but de proposer un modèle générique pour un produit communicant servant d'orientation pour la génération d'une proposition méthodologique pour concevoir des produits communicants. Dans la Figure 64 nous présentons le modèle proposé pour un produit communicant.

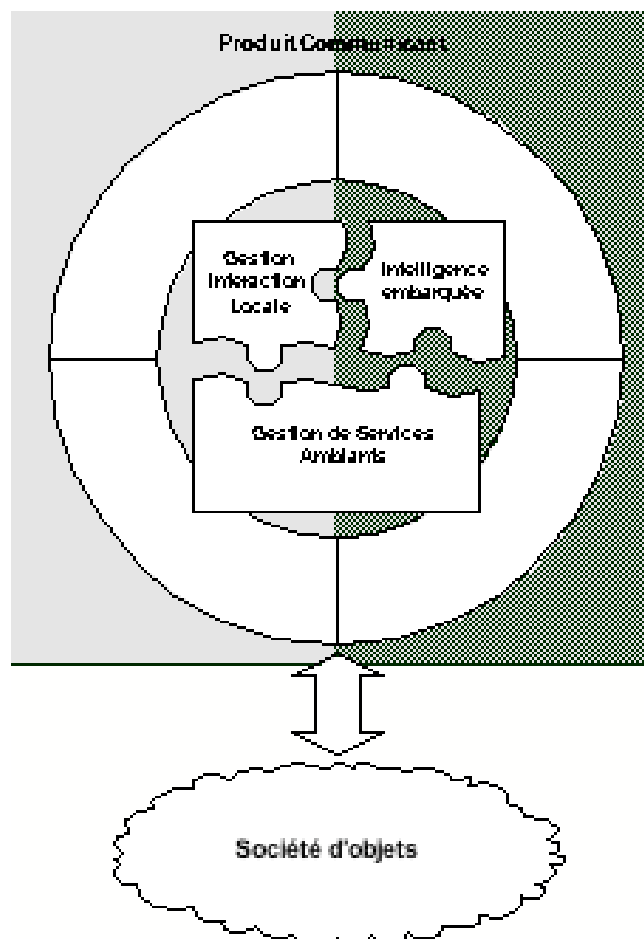


Figure 64 : Modèle conceptuel d'un produit communicant.

Dans cette conceptualisation un produit communicant est composé par :

- **Module de gestion d'interaction locale :** Le but de ce module est de permettre l'interaction entre un produit physique et des éléments de proximité immédiate du produit sans utiliser une infrastructure de services ambiants. C'est le domaine d'application des technologies RFID permettant de relier un produit physique avec une interface RFID (type PDA) afin de garantir au produit communicant d'offrir ses services aux utilisateurs en fonction de son profil, de sa localisation, et d'autres informations contextuelles.
- **Module de gestion de services ambiants :** Le but de ce module est de représenter les fonctionnalités de base qu'un produit communicant doit fournir dans un réseau de services ambiants. Les fonctionnalités sont : l'identification unique d'un produit physique dans un réseau de services ambiants, la localisation logique et géographique d'un produit physique dans le réseau de services ambiant, la découverte des services d'autres objets communicants (dispositifs machines, ...), la description de ses services afin de les faire apparaître dans le réseau ambiant pour les autres entités intéressées, l'invocation de services à distance sur d'autres objets communicants et la gestion des événements propres et d'autres entités interagissant avec le produit communicant. Ces fonctionnalités rendent possible l'interaction entre objets à distance au travers d'une communication filaire ou sans fil, permettant la conformation d'une société d'objets communicants.
- **Module d'intelligence embarquée :** Ce module a pour but de permettre d'apporter à un produit communicant des capacités d'analyse de données, d'évaluation d'actions, et de prise de décisions afin de gérer son comportement. Ce module peut être intégré dans le produit communicant ou peut être développé par une entité représentant la partie virtuelle du produit communicant. La capacité d'analyse peut notamment être enrichie par des capteurs associés au produit communicant. Ainsi, le produit communicant peut avoir un rôle sensitif vers son environnement.

Cette conceptualisation sera développée dans le chapitre 4, afin de proposer une méthodologie permettant la transformation d'un produit physique en produit communicant supporté par ses services de nature ambiante dans une infrastructure de services ambiants UPnP.

En particulier, l'intégration de la technologie RFID et l'architecture UPnP a été testée avec un démonstrateur de laboratoire afin de valider l'interopérabilité fonctionnelle entre ces technologies. Après avoir testé la faisabilité d'intégration, nous avons suivi le cadre méthodologique proposé dans le chapitre suivant, afin d'enrichir un produit physique avec des services ambiants dans le but d'obtenir un produit communicant. Dans le chapitre 5 de ce manuscrit, nous présentons les démonstrateurs de laboratoire.

3.7 Conclusions

Les concepts de services, d'infrastructure et de réseau ambiant traités dans ce chapitre permettent de situer les éléments de base de notre problématique afin de proposer une méthodologie qui puisse supporter les interactions entre des objets communicants et des processus dans un système industriel. Les fonctionnalités génériques d'une architecture de services ambiants que nous avons définis sont basées sur le concept de service ambiant et permettent d'orienter les exigences conceptuelles d'une approche de solution. L'identification automatique d'un objet communicant permet de faire apparaître l'objet dans un réseau ambiant et en même temps permet de configurer ses services disponibles. Ce sont les services d'un objet communicant qui permettront de supporter les interactions avec des ressources proches ou distantes. Dans ce cas, les produits sont des objets communicants dans son cycle de vie et pouvant avoir un double rôle : demandeur de services ou fournisseur de services au même titre qu'un équipement intelligent dans une infrastructure informatique évoluée.

Dans ce chapitre, nous avons étudié la notion d'informatique ambiante, de réseau ambiant et de services ambiants. Ici, l'idée sous-jacente est la considération de l'entourage des objets communicants, des ressources, ou en général, des entités informatiques. Nous avons identifié les fonctionnalités génériques d'une architecture de services ambiants dans le but de faire communiquer des objets communicants avec des ressources informatiques. Nous avons analysés les architectures de services ambiants les plus importantes à ce jour. Une analyse comparative a été faite afin d'identifier l'architecture de services ambiant la plus pertinente pour nos objectifs. Des critères de choix ont été identifiés afin de réaliser une synthèse comparative des architectures de services ambiants étudiées. Ces critères de base ont permis d'identifier l'architecture UPnP comme l'architecture de base qui sera intégrée dans notre méthodologie afin de supporter les interactions des objets communicants dans un contexte industriel et logistique.

4 Proposition Méthodologique

4.1 Présentation de l'approche proposée

Dans ce chapitre nous présenterons notre proposition méthodologique pour faire interagir un produit physique et sa représentation virtuelle avec des demandeurs et fournisseurs de services dans le contexte de la chaîne logistique. Notre proposition intègre les six fonctionnalités de base présentées dans le chapitre 3 appliquées à un produit communicant : l'identification automatique d'un produit communicant, la localisation logique et géographique d'un produit communicant, la découverte automatique des services d'un produit communicant, la description d'un produit communicant et ses services, l'invocation à distance des services d'un produit communicant, et la notification des événements initiés par un produit communicant. Ces fonctionnalités permettent le dialogue entre des demandeurs et des fournisseurs de services. Tous les acteurs de la chaîne logistique (objets communicants) doivent s'identifier automatiquement comme des demandeurs et/ou fournisseurs de services appartenant à un domaine de service déterminé. Comme nous l'avons dit auparavant, les services que chaque objet communicant met à disposition dans une infrastructure de services représentent le lien informationnel permettant les interactions entre demandeurs et fournisseurs de services. Chaque produit / objet communicant doit gérer ses services : les annoncer, les exécuter, les actualiser et les diffuser avec ses mises à jour quand elles sont nécessaires. Cet aspect dynamique est essentiel dans la gestion de services avancés que nous proposons.

4.1.1 Le produit physique et ses services dans son cycle de vie

Dans notre approche, nous considérons un produit comme un acteur qui gère son évolution en coopération avec les différents acteurs de la chaîne logistique. Dans ce sens, un produit est vu comme un objet dual, physique et virtuel [Cea et *al.*, 2004]. Un **produit physique** est un objet matériel caractérisé par des informations de nature intrinsèque telles que sa couleur, sa forme, sa taille, son poids, son volume... L'image virtuelle du produit, ou **produit virtuel**, ou bien contrepartie virtuelle [Römer et *al.*, 2004], est une extension informationnelle du produit physique accessible grâce à un ensemble de ressources physiques et à un système d'information distribué. En effet, le produit virtuel représente un produit augmenté [Falk, 1999] dans un réseau ambiant. Comme exemple d'image virtuelle d'un produit physique, nous citons sa nomenclature composante, sa gamme de fabrication, le lieu et conditions de stockage, son mode d'emploi, sa procédure de maintenance, sa procédure de recyclage, ... Le produit virtuel est aussi un conteneur de méthodes d'action et de décision permettant de lui donner une capacité de traitement de l'information, voire de décision, afin d'interagir avec son environnement. Concrètement, la liaison entre le produit physique et le produit virtuel peut être réalisée grâce à des technologies d'identification automatique et des systèmes ambiants.

Un **acteur de la chaîne logistique** est une entité qui prend part au cycle de vie d'un produit physique [Maxwell et Van der Vorst, 2003] représentant un **processus**, un **utilisateur** ou un autre **produit**. Cette entité peut effectuer une demande de service et/ou répondre aux demandes de services effectuées par un produit ou d'un autre acteur de la chaîne logistique. Comme exemple d'acteurs de la chaîne logistique, nous citons un processus de production, un processus d'entreposage, un opérateur dans un processus de contrôle, ou un produit physique qui est en relation avec des processus, des utilisateurs, et même, avec d'autres produits dans un endroit déterminé. Chaque acteur de la chaîne logistique a un profil associé. Le **profil de l'acteur de la chaîne logistique** permet de caractériser un acteur au moyen d'un ensemble d'informations (identification, compétences, responsabilités, privilèges,

autorisations, ...) avec le but de configurer son comportement dans la chaîne logistique. Dans notre approche, un **processus** représente un ensemble d'opérations permettant à un acteur de la chaîne logistique d'effectuer un traitement approprié de l'information, d'exécuter des actions sur un autre acteur et de prendre des décisions concernant son cycle de vie. Le **cycle de vie** d'un produit physique est composé d'un ensemble de processus associés à chaque phase de son cycle de vie.

Une **interaction** représente une influence réciproque entre deux acteurs de la chaîne logistique. Dans ce sens, une interaction peut être initiée par un produit physique ou par un autre acteur de la chaîne logistique. Par exemple, par rapport aux interactions qu'un acteur de la chaîne logistique peut réaliser sur un produit physique, nous citons : une demande d'information, une manipulation ou une transformation. Dans le cas d'une interaction initiée par le produit, il peut s'agir d'une auto-déclaration, d'une demande d'intervention sur un acteur de la chaîne logistique,... . En plus, les interactions peuvent se dérouler entre produits.

Le concept de service a été traité comme élément central par plusieurs auteurs [O'Sullivan et al., 2002] [Figue, 2004] [Gu et al., 2005] [Console et al., 2003] [Wang et al., 2004] [Ciminiera et al., 2004] [Georgakopoulos et al., 2002]. Voir le point 2.2.4 de ce document afin de comprendre comment ces auteurs interprètent le concept de service. Pour nous, un **service** représente une fonctionnalité permettant à un acteur de la chaîne logistique d'invoquer l'exécution d'une ou plusieurs actions à distance afin d'obtenir une réponse apportant des éléments de décision à l'entité qu'initie la communication. Les interactions entre les acteurs de la chaîne logistique sont basées sur le concept de service [Cea et Bajic, 2004]. Ainsi, une interaction entre deux acteurs – dont un produit – est réalisée par l'invocation d'un service sur le produit ou sur le processus, dans l'objectif d'obtenir une fonctionnalité offerte par le fournisseur de services. Dans notre approche, nous considérons qu'un produit dans son cycle de vie peut être représenté comme un objet communicant.

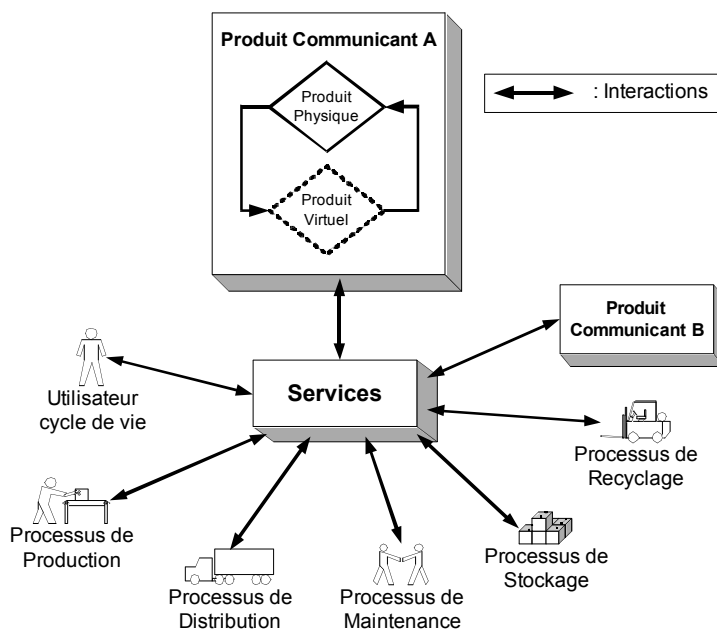


Figure 65 : Interactions entre un produit communicant, des processus et des utilisateurs au travers des services dans la chaîne logistique.

La Figure 65 illustre les interactions, représentées par les flèches, entre un produit communicant A, intégrant une partie physique et une partie virtuelle, et des processus associés à des phases du cycle de vie d'un produit (production, distribution, maintenance, stockage, recyclage, ...) dans la chaîne logistique au travers des services. Aussi, nous représentons un utilisateur interagissant avec les

produits communicants A et B, et des processus (production, distribution, ...) via les services. Il est possible aussi que les produits communicants A et B puissent interagir entre eux par les services. Dans puisement la figure nous mettons en évidence que les acteurs de la chaîne logistiques sont des produits / objets communicants, des processus, et des utilisateurs interagissant entre eux en passant par les services.

Dans ce sens, selon l'analyse que nous avons réalisée dans le chapitre 2, un produit / objet communicant doit être capable de : posséder une identification unique ; communiquer avec son environnement proche et distant ; mémoriser et gérer des informations propres ; disposer d'un langage de dialogue ; participer aux processus de décision durant son cycle de vie ; surveiller et contrôler son environnement ; offrir des services sur son cycle de vie ; demander des services aux acteurs participant à son cycle de vie. Ceci implique de caractériser un produit physique comme un objet dual, qui peut être, soit un fournisseur des services, soit un demandeur de services. Dans ce chapitre nous montrerons comment toutes ces capacités peuvent être accomplies grâce à notre proposition méthodologique.

4.2 Proposition méthodologique : types d'objets communicants

Notre proposition méthodologique pour créer et gérer des objets communicants, dans le domaine de la chaîne logistique, suggère une intégration effective entre un produit physique matériel et sa représentation virtuelle. Nous utilisons la technologie RFID [Paret, 2001] [Paret, 2003] [Finkenzeller, 2003] comme technologie « Auto ID » pour identifier automatiquement un produit physique dans un réseau ambiant. La proposition est suffisamment flexible pour intégrer une autre technologie d'identification automatique afin d'implémenter la notion de service ambiant. Comme nous l'avons précisé dans le chapitre 3, nous avons adopté comme cadre de référence l'architecture de service ambiant UPnP [UPnP Forum, 2003] pour développer les fonctionnalités de base afin de faire interagir les objets communicants avec les acteurs de la chaîne logistique.

Dans notre méthodologie, nous modélisons un objet communicant comme un objet dual, composé d'une partie physique matériel et d'une partie informationnelle, doté de capacités de mémorisation, de perception, de communication, d'action et de décision. Dans les points suivants nous présenterons les trois cas d'analyse que nous avons considéré, du plus simple au plus complexe.

4.2.1 Cas 1 : Objet communicant passif

Le premier cas est représenté par l'ensemble de deux éléments : un produit physique portant une étiquette électronique (produit tagué) et un dispositif UPnP doté de capacités de communication RFID. Ce que nous voulons créer avec cet ensemble, c'est un objet communicant agissant comme un fournisseur de services [Cea et Bajic, 2005]. En effet, l'ensemble (produit - dispositif UPnP) permet d'offrir des services aux acteurs de la chaîne logistique. Dans ce cas, ce type d'objet communicant a un rôle passif puisqu'il peut seulement répondre aux requêtes des demandeurs de services sans faire de requêtes vers l'extérieur (environnement) de sa propre initiative. Ainsi, le produit est transformé en un objet que nous appelons objet communicant passif. Plus précisément, les composants physiques de cet objet communicant passif sont les suivants : un produit physique, une étiquette électronique, une image de l'objet dans une base de données et un ordinateur fixe ou mobile agissant comme dispositif UPnP avec un système RFID intégré. L'étiquette électronique de l'objet physique peut avoir des capteurs intégrés afin d'équiper l'objet de capacités sensorielles [Beigl et *al.*, 2004] telles que température, humidité, lumière, mouvement, proximité, force / pression, Cela permet de caractériser cet objet comme un objet communicant passif sensitif.

La Figure 66 schématise ce premier cas formant un objet communicant passif. On observe dans la figure le produit physique tagué étant identifié par l'interface RFID du dispositif UPnP dans une zone d'attraction RFID. En fonction des caractéristiques du système RFID à utiliser, la zone d'attraction aura une forme et une extension particulière. On observe également dans le dispositif UPnP ses fonctionnalités d'adressage automatique, de découverte automatique des autres dispositifs, de description des services offerts par l'objet communicant, de contrôle afin d'être contrôlé à distance et de gestion d'événements souscrits par d'autres dispositifs UPnP. La figure illustre aussi une extension virtuelle de l'objet communicant présente dans une base de données accessible via Internet. Dans la partie inférieure de la figure, on observe un point de contrôle (et ses fonctionnalités) capable d'interagir avec l'objet communicant à distance. Le point de contrôle représente un acteur de la chaîne logistique.

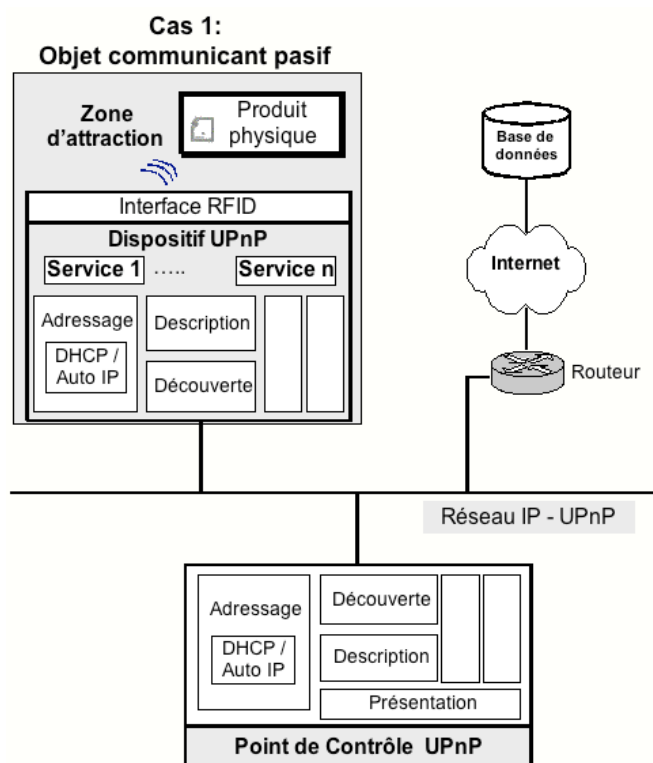


Figure 66 : Objet communicant passif dans un réseau ambiant UPnP.

Lorsqu'un produit physique arrive dans un domaine de services, le produit est automatiquement identifié dans l'infrastructure de services au moyen d'un identifiant unique stocké dans son étiquette électronique. A ce moment, l'objet communicant passif, créé automatiquement, est disponible dans le réseau ambiant pour offrir ses services codifiés en totalité ou partiellement dans la mémoire du produit. Ces services, dont les descriptions exhaustives sont contenues dans des fichiers XML [Jeronimo et Weast, 2003], sont interprétés et installés dans la mémoire du dispositif UPnP. Chaque description de services en XML décrit les actions offertes par l'objet communicant et ses variables d'état. Ainsi, les points de contrôle intéressés peuvent interroger le produit physique afin de connaître ses fonctionnalités offertes. Les services offerts peuvent être obtenus in situ, en lisant les données stockées sur l'étiquette électronique, ou obtenus de manière distante, en accédant à une base de données afin d'obtenir l'information spécialisée, apportée par exemple par le fabricant du produit (manuel d'usage, caractéristiques techniques, résultat de contrôle de qualité, ...). Les services du produit physique représentent le lien avec son image virtuelle. Les points de contrôle peuvent

souscrire aux événements d'un objet communicant passif pour être informés au changement d'une variable d'état.

L'objet communicant passif, conformément aux potentialités d'UPnP, obtient une adresse unique dans le réseau ambiant afin d'interagir avec d'autres objets communicants et avec des points de contrôle. La localisation de l'objet communicant peut directe ou indirecte. Pour le mécanisme direct, nous proposons d'attacher un système de géo-positionnement [Theiss et al., 2005] aux objets communicants afin de connaître leur localisation géographique. Par exemple, un système GPS (Global Positioning System) permet de localiser des objets sur la surface terrestre avec une exactitude d'environ 30 mètres en utilisant un réseau de satellites et un dispositif récepteur de signaux standard [DOF-USA, 2001]. Un système DGPS (GPS Différentiel) permet de réduire les sources d'erreur afin de gagner en exactitude [Omae et al., 2001] [Casademont et al., 2004]. Dans le but de comparer les systèmes GPS et DGPS, nous citons l'étude expérimentale développée par [Yoshimura et Hasegawa, 2003]. Ils ont comparé les performances en terme de précision⁴³ et d'exactitude⁴⁴ d'un système GPS et d'un système DGPS dans différentes conditions dans une forêt. Ils ont trouvé qu'un système DGPS atteindrait une exactitude plus élevée qu'un système GPS et des niveaux de précision semblables dans les scénarios étudiés.

Dans le cas de la radiofréquence, le principe de triangulation [Château et al., 2003] peut être employé pour localiser un produit portant une étiquette électronique. Cette dernière possibilité demande, au minimum, de disposer de trois lecteurs RFID dans une zone de lecture commune où se trouve l'objet afin d'estimer sa localisation. La Figure 67 illustre le principe de triangulation appliqué à la localisation des objets communicants. On observe, à gauche, trois lecteurs (1, 2 et 3) identifiant l'objet A. Il est possible d'estimer la distance entre un lecteur RFID (1, 2, 3) et l'objet en utilisant le temps de voyage du signal électromagnétique ou le niveau de bruit présent dans le signal [Fishin et al., 2004]. Chaque distance (d1, d2, d3) représente le rayon d'une sphère autour d'un lecteur RFID. L'intersection des trois sphères détermine la localisation de l'objet A. Dans une distribution symétrique de lecteurs équidistants à une distance « a », la localisation (x,y,z) de l'objet est obtenue selon la formule indiquée à droite de la Figure 67.

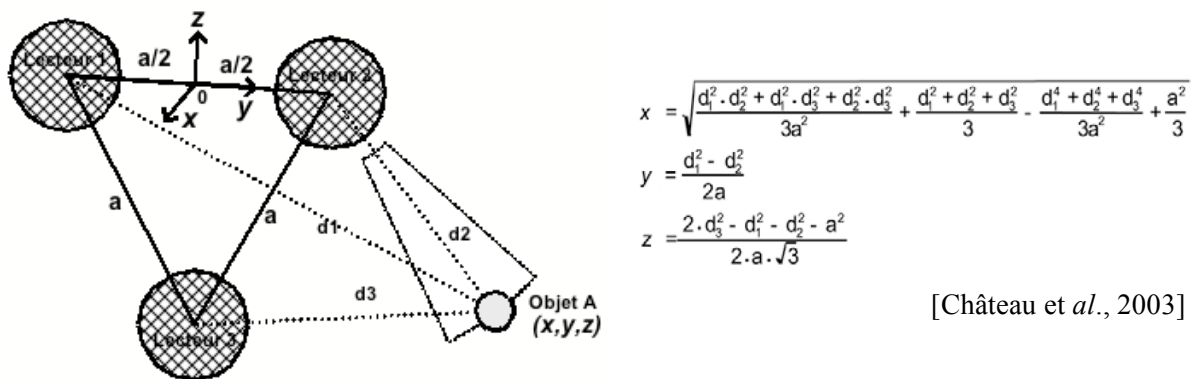


Figure 67 : Principe de triangulation appliqué à la localisation des objets communicants.

⁴³ La précision est définie comme la racine carrée de la somme des variances observées entre les valeurs obtenues et la moyenne arithmétique de ces valeurs pour les axes « x », « y » et « z » dans chaque expérimentation. C'est-à-dire, la précision montre le degré de dispersion des données observées.

⁴⁴ L'exactitude est définie comme la racine carrée de la somme des variances entre les valeurs obtenues et les valeurs attendues (vrais) pour les axes « x », « y » et « z » dans chaque expérimentation. C'est-à-dire, l'exactitude montre la dispersion des valeurs obtenues par rapport aux valeurs réelles.

Afin d'obtenir des résultats satisfaisants avec cette méthode de localisation, on doit utiliser des étiquettes actives opérant à de hautes fréquences pour amplifier au maximum la zone d'attraction. La notion de distance entre le lecteur et l'étiquette électronique peut être utilisée, comme mécanisme de sécurité, afin d'éliminer de possibles écoutes pirates et les autorisant seulement aux demandeurs de services qui sont situés dans un périmètre permis. Alternativement, comme moyen indirect de localisation des produits physiques, nous proposons d'enregistrer sur l'étiquette électronique un code de localisation afin de stocker la dernière localisation connue du produit et de la modifier si le produit est transporté par un acteur de la chaîne logistique. Dans ce cas, le produit aura un code actualisé de sa localisation chaque fois qu'il arrive à une nouvelle zone de travail. Pour un tel effet, il faut que le système RFID ait des capacités de lecture et d'écriture.

La découverte de nouveaux produits supporte le mécanisme d'annonce automatique lorsque le produit physique est identifié dans une zone d'attraction RFID. De plus, la découverte supporte le mécanisme de recherche fait par les objets communicants lorsqu'ils sont arrivés au réseau après l'arrivée de l'objet communicant passif. La fonctionnalité de contrôle supporte l'invocation des actions à distance par des points de contrôles ou des objets communicants demandeurs de services. La gestion d'événements permet aux points de contrôle d'être actualisés à propos des possibles changements des variables de l'objet communicant.

Cette approche permet d'assigner aux acteurs de la chaîne logistique des fonctionnalités d'un point de contrôle, c'est-à-dire, de demandeur de services vers les objets communicants fournisseurs de services. Les acteurs de la chaîne logistique peuvent demander des services au long du cycle de vie d'un objet communicant. Après la fin de la vie « réelle » d'un produit physique, éventuellement sa vie informationnelle (produit virtuel) peut continuer un certain temps. Dans le domaine de la fabrication, nous pouvons mentionner comme exemple d'acteurs de la chaîne logistique : des machines, des robots, des automates programmables, des chariots de transport, des opérateurs équipés avec des (mini) ordinateurs portables, des systèmes de supervision et de contrôle, des objets communicants, Ainsi, ces acteurs, équipés avec des capacités de communication UPnP filaire ou sans fil, peuvent interagir avec l'objet communicant passif à distance. Comme exemple de produit physique agissant en tant qu'objet communicant passif, nous citons les produits à faible valeur, tels que : des produits de commercialisation, des boîtes d'ensembles de produits physiques, des palettes groupant des boîtes, des cartons, etc. Ainsi, les acteurs de la chaîne logistique disposeront des services des objets communicants au moment où ils sont identifiés dans un domaine de services.

L'objet communicant passif proposé, grâce à ses capacités de mémorisation, de perception de communication, d'action et de décision, est un véritable objet communicant. En fait, l'objet communicant est seulement passif vers l'extérieur et actif à l'intérieur. C'est-à-dire, l'objet peut exécuter des opérations à l'intérieur et peut prendre de décisions simples en utilisant son information interne. Ceci implique que l'objet communicant est capable d'exécuter des services sur lui-même et d'évaluer ses résultats. L'objet communicant passif peut participer activement dans son cycle de vie en apportant des services pour faciliter la prise de décisions dans tous les contextes où il est identifié et traité. Comme avantage de ce cas, nous citons le bas coût de son implémentation, la flexibilité dans la formulation de services métiers à proposer pour chaque situation particulière, et la rapide intégration dans un système industriel en utilisant les ressources informatiques actuelles enrichies avec des équipements RFID. Mais le comportement actif vers l'environnement est absent dans cette conceptualisation. Le deuxième cas proposé permet de résoudre cette limitation ou obstacle.

4.2.2 Cas 2 : Objet communicant actif

Dans ce deuxième cas, nous représentons un produit physique en communication avec un dispositif UPnP au moyen d'une étiquette électronique ayant pour but de créer une entité que nous appelons objet communicant actif. La nature active de l'objet lui permet de demander des services à des acteurs de la chaîne logistique. En plus, il peut fournir des services dans un réseau ambiant. Ainsi, l'objet communicant actif est un fournisseur et un demandeur de services. Les composants de notre deuxième cas proposé sont : un produit physique équipé d'une étiquette électronique, une image virtuelle du produit disponible dans une base de données, un ordinateur fixe ou mobile avec un système RFID intégré agissant comme dispositif UPnP incorporant les fonctionnalités d'un point de contrôle (Dispositif (cp) UPnP : définition dans la section 3.8.3). Le Dispositif (cp) UPnP intègre des capacités de communication RFID afin de permettre l'identification automatique d'un produit entrant dans une zone d'attraction RFID. Nous considérons l'objet communicant actif plus intelligent que l'objet communicant passif décrit dans le point précédent, du fait qu'il est capable de demander des services (comportement actif) et pouvant, avec les réponses obtenues, évaluer ses options afin de prendre des décisions adaptées à ses caractéristiques constitutives [Cea et Bajic, 2005]. Pour supporter ces opérations d'évaluation et de décision, l'objet communicant actif contient une couche logicielle intégrée dans le dispositif (cp) UPnP.

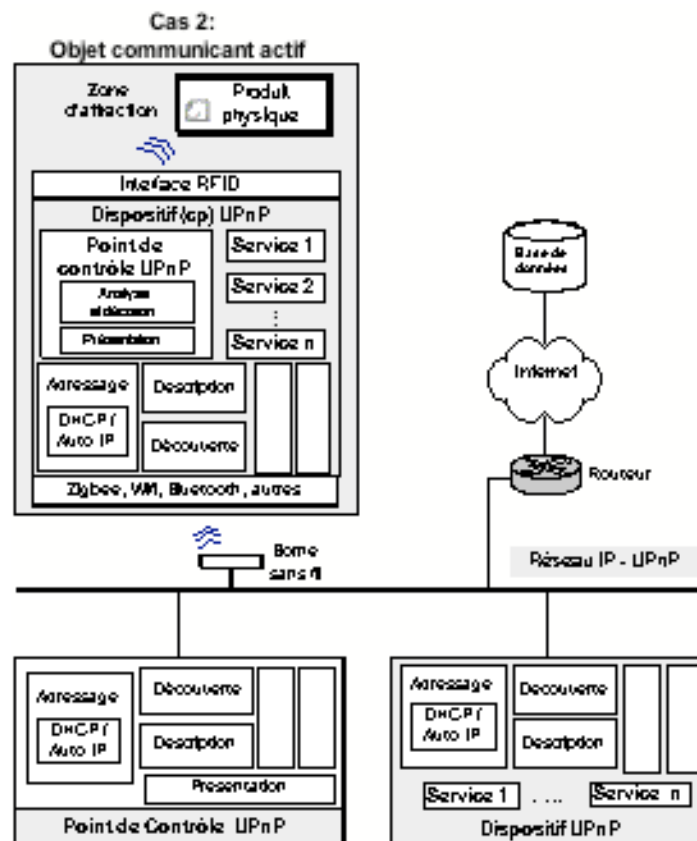


Figure 68 : Objet communicant actif dans un réseau ambiant UPnP.

Dans la Figure 68, nous représentons l'objet communicant actif intégrant ses fonctionnalités de dispositif UPnP et de point de contrôle UPnP. Spécifiquement, nous intégrons dans ce dernier une **couche d'analyse et de décision** pour supporter les capacités d'action et de décision attendues de l'objet communicant. Ceci implique que l'objet communicant, après l'exécution des services vers des acteurs de la chaîne logistique, peut analyser les résultats et prendre des décisions. L'image virtuelle

de l'objet sur une base de données est accessible au travers des services offerts par le produit. Dans la partie inférieure de la Figure 68 nous représentons un point de contrôle (demandeur de services), ou un acteur de la chaîne logistique, qui interagit avec l'objet communicant ou avec un autre dispositif UPnP. La figure également incorpore un dispositif UPnP (fournisseur de services) qui peut être interrogé par l'objet communicant actif ou un acteur de la chaîne logistique.

L'objet communicant actif hérite de toutes les fonctionnalités décrites pour l'objet communicant passif. En plus, l'objet communicant actif développe des fonctionnalités pour être un demandeur de services. Pour cela, l'objet communicant peut visualiser tous les acteurs de la chaîne logistique présents dans le réseau ambiant, afin de choisir, interroger et contrôler ceux qui sont importants pour accomplir ses tâches. Il peut aussi souscrire aux événements produits par des acteurs de la chaîne logistique afin de surveiller son environnement et d'être informé de tout changement qui peut affecter son état d'équilibre. Par exemple, dans la phase de production, l'objet communicant actif peut demander des services vers des machines, des robots, des automates, ... afin de planifier, négocier, exécuter et suivre le déroulement de ses opérations en fonction de ses paramètres de qualité, du temps de réponse attendue, du coût, entre autres facteurs, définis pour le client au moment de la commande. Dans ce cas, le produit est le « porte-parole » du client dans les processus de production. Ceci représente une application du concept de personnalisation des produits [Da Silveira et al., 2001] [Agard et Tollenaere, 2002]. Si un produit ne peut pas négocier de manière satisfaisante ses opérations, il peut demander l'aide d'un système négociateur, ou même, communiquer avec d'autres acteurs en dehors du système de production.

Un autre scénario d'interaction est la communication entre produits. Dans ce cas, par exemple un objet communicant actif peut interagir avec un objet communicant passif pour lui demander des services : changer la place dans une file d'attente, arrêter une opération momentanément, ou simplement, parce que les objets communicants feront partie d'un nouvel objet communicant. Dans ce dernier cas, les objets pourront communiquer pour vérifier l'appairage, ou autoriser un assemblage correct. Ultérieurement, la phase inverse nécessite la gestion d'une éventuelle réutilisation de certaines parties – *end of life product management* [Parlikad et McFarlane, 2004] - afin de conformer des objets reconditionnés par exemple.

Comme avantage de ce deuxième cas proposé, nous pouvons citer la flexibilité d'intégration de cette solution dans des infrastructures industrielles actuelles. Il faut préciser que le comportement actif implique de bien connaître tout le processus touchant le produit physique afin de bien paramétrer la couche logiciel du dispositif (cp) UPnP afin de définir des interactions automatiques entre les acteurs de la chaîne logistique. Pour cela, nous proposons de réaliser une étude des besoins pour chaque domaine de la chaîne logistique afin de définir les processus, les utilisateurs (clients internes et externes), les ressources physiques et informationnelles, les paramètres d'opération, les variables de comportement, les états des éléments conformant les processus, les activités internes des processus, les relations avec l'environnement, et les règles d'opération et de décision. Comme outil de modélisation de ces interactions, nous proposons d'utiliser le langage de modélisation UML 2.0 (*Unified Modeling Language*) [OMG UML, 2004] [Bauer et Odell, 2005] [Willard, 2006].

4.2.3 Cas 3 : Objet communicant actif unifié

Le troisième cas, l'objet communicant actif unifié, est représenté par un dispositif (cp) UPnP complètement intégré dans le produit physique ou vice-versa. Cela permet de caractériser l'objet communicant actif unifié résultant comme un demandeur et fournisseur de services dans son cycle de vie [Cea et Bajic, 2005]. Ce type d'objet communicant actif hérite les fonctionnalités de gestion de

services ambiants des deux cas précédentes. Particulièrement, dans ce cas, l'objet n'a plus besoin d'être identifié au travers de la technologie RFID. Sinon, l'objet est identifié automatiquement grâce à un code stocké dans la mémoire de l'objet communicant. Ce cas de solution est le modèle le plus complexe proposé dans notre démarche méthodologique. Dans la partie supérieure de la Figure 69 nous représentons le produit physique en intégrant un dispositif (cp) UPnP connecté à un réseau ambiant UPnP via une communication sans fil. Le produit physique peut avoir des capteurs et des actionneurs intégrés afin d'interagir directement avec son environnement. Ainsi, l'objet actif devient un objet communicant sensitif. Ceci permet d'implémenter la capacité de perception et d'action sur l'objet communicant. Dans la figure, nous représentons un acteur de la chaîne logistique (point de contrôle UPnP) ayant la possibilité de demander des services à l'objet communicant actif unifié. Egalement, l'objet communicant peut initier des actions vers d'autres acteurs de la chaîne logistique étant des fournisseurs de services. La capacité d'action est renforcée avec des traitements autonomes sur l'objet communicant. Après que l'objet a reçu des réponses à ses demandes, il prend des décisions afin d'évaluer ses actions futures. En plus, l'objet communicant actif unifié peut être augmenté avec des services accessibles sur une ou plusieurs bases de données distantes.

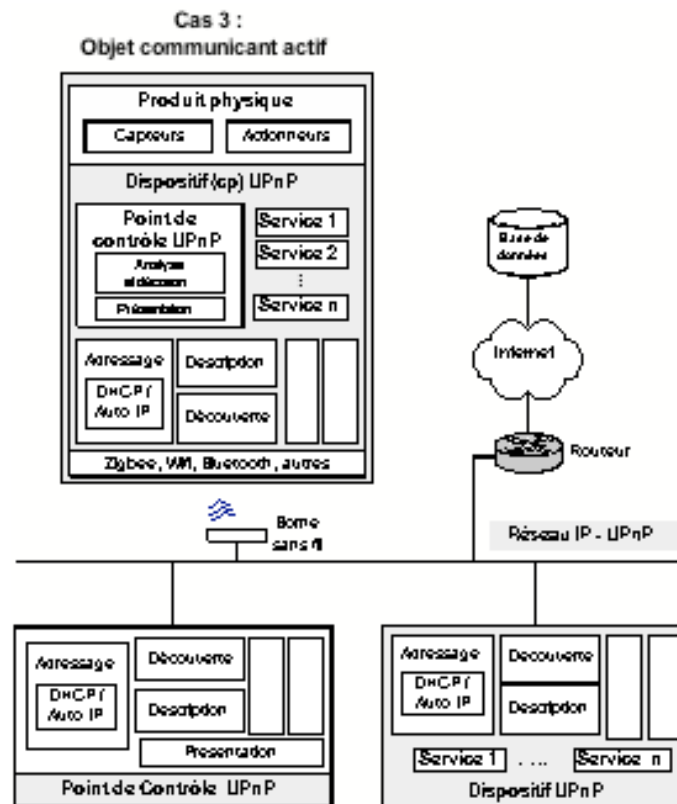


Figure 69 : Objet communicant actif unifié

L'objet communicant actif unifié représente l'union du produit physique avec la technologie microélectronique requise pour conformer un objet communicant « homogène » dans un même corps géométrique. Dans ce cas, la forme du produit physique absorbe de manière transparente la technologie qui l'enrichit. Nous pouvons citer comme exemples des objets communicants actifs unifiés des produits de consommation finale transformés en produits UPnP afin de créer de communautés virtuelles de produits dans un endroit donné. On imagine des produits communiquent entre eux dans des espaces publics (centres commerciaux, bibliothèques, bus, train, ...) et des espaces privés (maison, industrie, banque, ...) afin de demander / offrir des services tels que la sécurité industrielle, la surveillance à distance, la réception des événements adaptés au contexte (alarmes, avertissements, suggestions, ..), la mise à jour des objets, etc. Ici, l'importance est dans le

comportement qui émerge des interactions entre les objets. Etant donné la complexité de ces produits, nous envisageons des interactions par les services entre ces objets après leurs fabrication, c'est-à-dire, dans les fonctions de transport, de stockage, d'usage, de maintenance et de recyclage. Par exemple, l'autodiagnostic et l'auto-maintenance sont des fonctions qu'il est parfaitement possible à réaliser en communication à distance via Internet avec les fournisseurs. En plus, l'objet pourrait effectuer automatiquement des commandes électroniques, via des services Web, pour acquérir des objets complémentaires (des batteries, de nouvelles parties, des nouvelles fonctionnalités à installer ...) afin de gérer son cycle de vie correctement. Dans ce cas, nous pouvons parler d'actions pilotées par l'objet en fonction de son état. Mais également, les capacités de perception des objets peuvent être étendues au travers d'un réseau de capteurs filaires ou sans fil dans le but d'obtenir à distance des informations sur l'environnement physique afin de filtrer, agréger, analyser, sauvegarder et réagir aux mesures acquises [Desertot et *al.*, 2005]. Particulièrement, la problématique dans le domaine des réseaux de capteurs sans fil est associée à l'optimisation de la communication entre les nœuds conformant des configurations topologiques dynamiques en utilisant une consommation d'énergie minimale pour différents scénarios d'utilisation [Akyildiz, 2002] [Wang et *al.*, 2006].

Comme inconvénient à cette solution, nous citons le coût d'implémentation relativement plus élevé par rapport aux deux cas précédents. Cependant, les nouvelles possibilités d'interactions et de création de nouveaux services justifient cette augmentation. Les sources d'énergie pour ce type d'objet communicant peuvent être de nature limitée (des batteries, ...) ou de nature permanente. Le type d'alimentation sera défini en accord avec la finalité de l'objet. Dans certains cas, la vie utile de l'objet peut coïncider avec la vie de la source d'alimentation.

4.3 Modélisation des services pour un objet communicant

La modélisation des services permet de formaliser les relations entre les objets communicants et les acteurs de la chaîne logistique. Il est nécessaire de spécifier les exigences minimales de ce processus de modélisation afin de bien guider les concepteurs d'un système ambiant géré par des objets communicants. Ces orientations permettent de spécifier les degrés de liberté souhaités envers les possibilités d'usages des services d'un objet communicant. Ces degrés de liberté sont déterminés en fonction du contexte de l'objet communicant. Nous avons identifié trois facteurs contextuels déterminant les interactions entre les objets communicants et les acteurs de la chaîne logistique :

- la nature de l'objet communicant ;
- le domaine d'interaction des objets communicants ;
- les profils des acteurs de la chaîne logistique.

4.3.1 Facteurs déterminant les services des objets communicants

4.3.1.1 Nature de l'objet physique

La nature de l'objet physique est représentée par un ensemble de caractéristiques intrinsèques associées à l'objet physique. Ce sont ces caractéristiques ou variables qui permettent de configurer une typologie d'objets. Afin de grouper les objets par catégories, nous utilisons le concept de famille d'objets. Si l'objet est un produit, nous parlons de famille de produits.

Pour [Jiao et Tseng, 1999] une famille de produits est un ensemble de produits qui sont définis en s'appuyant sur trois points de vue : fonctionnel, technique et fabrication. C'est à dire, la demande d'un

client spécifiée en termes de fonctionnalités attendues d'un objet doit être technologiquement faisable et « fabricable » par un système de production. Selon [Zhang et *al.*, 2005] une famille de produits est un ensemble de produits connexes fabriqués par une plateforme commune afin d'atteindre une haute variété externe de produits (visibilité) avec un variété interne minimale (système).

Nous considérons qu'une famille d'objets ou de produits est définie comme un ensemble d'objets qui partagent une même finalité. En conséquent, ils ont des caractéristiques de base communes (attributs). Dans ce sens, un objet particulier est un cas spécial d'objet générique. L'objet particulier intègre moins de variables (variabilité) que l'objet générique. Comme critère de décomposition des familles, nous proposons d'utiliser les attributs fonctionnels caractérisant un objet physique. Le numéro de familles à considérer peut être déterminé au moyen d'un modèle d'optimisation mathématique en considérant comme variables significatives : la demande de variété de produits par les clients, la variété de produits à fabriquer, le numéro de sites de production et ses caractéristiques (flux, stock, capacité,...) le temps d'opération (production, stockage, transport, ...) et le coût total de gestion de la variété dans la chaîne logistique. Par exemple [Lamothe et *al.*, 2006] proposent un modèle de programmation linéaire pour trouver le numéro de variants (instances) d'une famille de produits et la configuration de la chaîne logistique globale afin de minimiser le coût total.

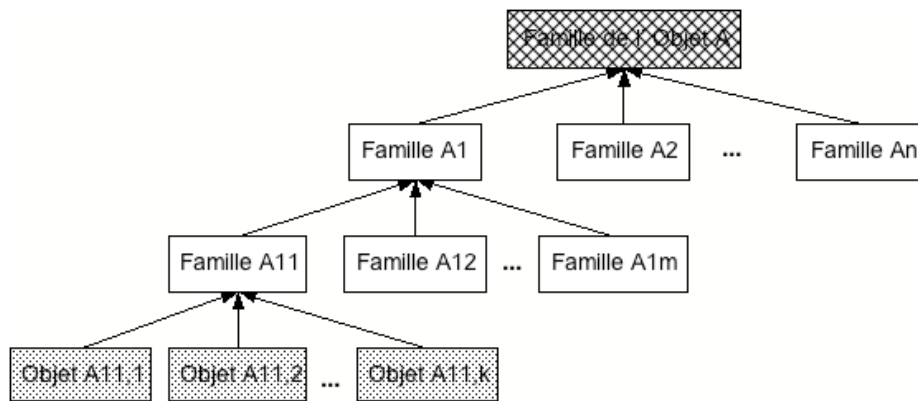


Figure 70 : Exemple de décomposition de familles d'un objet

La Figure 70 illustre la décomposition de familles de l'objet A. L'objet A est associé à une famille racine qui est composée par plusieurs sous-familles (A1, A2, ... An). Chaque sous-famille est un cas particulier de la famille d'origine. Dans ce sens, la Famille A11 est une sous-famille de la Famille A1 qui est à la fois une sous-famille de la famille racine de l'objet A. Particulièrement, l'objet A11,1 est une instance spécifique de la famille A11. Ainsi, chaque instance spécifique associée à une famille racine représente un ensemble de services possibles pour ce type d'objet. Ces services peuvent avoir une finalité informative et/ou sensitive. Dans ce dernier cas, les services seront obtenus à partir des mesures réalisées par les capteurs liés (ou directement associés) à l'objet physique. L'objet informe ses caractéristiques internes (taille, forme, poids, température, humidité, etc.) et ses caractéristiques opérationnelles (restrictions d'usage, valeurs critiques des variables, ...) aux demandeurs de services afin de gérer son intégrité, comme par exemple : des caractéristiques décrivant son usage correct tels que valeurs maximales des variables critiques de l'objet. Éventuellement, de nouveaux services peuvent être créés sur la base des caractéristiques internes. Par exemple l'objet peut intégrer un mécanisme de comparaison entre une valeur externe d'un processus et la valeur d'un paramètre donné, afin de surveiller son environnement et de réagir opportunément.

4.3.1.2 Domaines d'interaction des objets communicants

Le concept de domaine d'interaction d'un objet communicant est en relation directe avec son domaine de services géographiques [Loke et al., 2005] à un instant donné pour un état quelconque de cet objet. Ainsi, le domaine d'interaction est défini par l'intersection de trois éléments : l'espace (endroit où se trouve l'objet), la forme ou l'état de l'objet (condition caractérisant l'objet physique) et le temps (le moment auquel l'objet est présent dans un réseau ambiant). Dans la chaîne logistique, nous identifions les espaces d'un objet comme un endroit physique associé aux phases du cycle de vie de l'objet physique. La forme adoptée par l'objet physique peut être une représentation physique ou virtuelle. Le temps permet de considérer la dimension temporelle des interactions afin de tracer l'histoire de l'objet.

Dans la Figure 71 nous illustrons les trois vues : l'espace représenté par E_1, E_2, \dots, E_m , la forme représentée par F_1, F_2, \dots, F_n et le temps représenté par t_1, t_2, \dots, t_k . L'intersection de ces trois vues configure le domaine d'interaction représenté par (E_3, F_3, t_i) . L'application de ce cadre de référence à la chaîne logistique, nous amène à définir l'espace (phases du cycle de vie d'un produit physique) comme: espace production, espace transport, espace distribution, espace stockage, espace usage, espace maintenance et espace recyclage. Comme formes ou états possibles d'un produit nous pouvons considérer : la conception initiale d'un produit, le produit en état semi-fini, le produit en état fini, le produit en état d'assemblage, le produit en état endommagé, le produit en état de destruction.

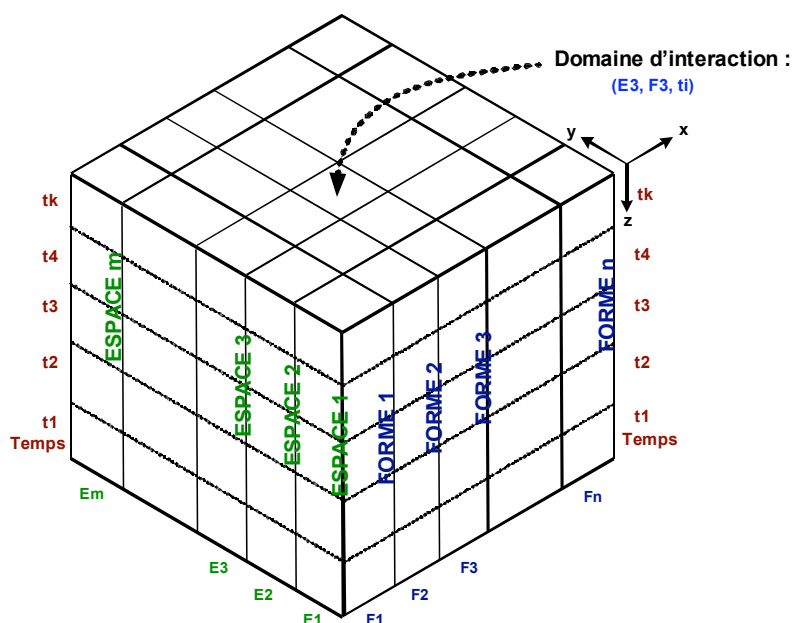


Figure 71: Domaines d'interactions pour un objet communicant.

Le domaine d'interaction permet de préciser l'information significative à analyser, extraire, grouper, manipuler, traiter, stocker, ... par rapport à un objet communicant. Les domaines d'interaction d'un objet communicant conditionnent les services qui pourront être proposés aux acteurs de la chaîne logistique. Ces services doivent s'adapter dynamiquement en fonction des déplacements et des changements d'état des objets communicants.

4.3.1.3 Profil d'acteur de la chaîne logistique

Le profil d'un acteur permet de configurer les possibilités d'interaction entre un acteur / utilisateur quelconque et un système informatique ambiant [Ciminiera et al., 2004] [Figue, 2004] [Bilchev et al., 2005] [Mokhtari et al., 2005]. En effet, les interactions entre les acteurs de la chaîne logistique et les objets communicants seront déterminées par le profil de l'acteur. Par acteur de la chaîne logistique, nous entendons un processus, un utilisateur ou un produit quelconque ayant les capacités de communication nécessaires pour interagir avec les objets communicants. Chaque acteur est associé à un ou plusieurs rôles dans un domaine de services fonctionnels. On constate qu'un domaine de services fonctionnels peut être associé à un ou plusieurs domaines de services géographiques. Chaque rôle particulier représente une fonction spécifique qui est associée à un domaine d'interaction à tout moment. Un rôle est décrit en termes d'un ensemble de processus. Des exemples d'acteurs de chaîne logistique sont : des humains, des machines, des robots, des produits, etc. Ces types d'acteurs peuvent se désagréger encore en fonction du rôle particulier de chacun. Par exemple : une personne peut avoir un rôle d'administrateur, opérateur, usager, ...

La Figure 72 illustre les rôles et les processus développés par un acteur de chaîne logistique (moyen de transport) afin de transporter des objets communicants. On observe que le moyen de transport a trois rôles : un rôle de transporteur des objets, un rôle de conservateur de l'intégrité des objets (température, humidité, ...) et un rôle de communicateur des états des objets communicants vers les acteurs externes intéressés. Dans ce dernier cas, le moyen de transport est un transmetteur et récepteur d'informations. Chaque rôle peut être associé à un ou plusieurs processus permettant d'accomplir une finalité donnée.

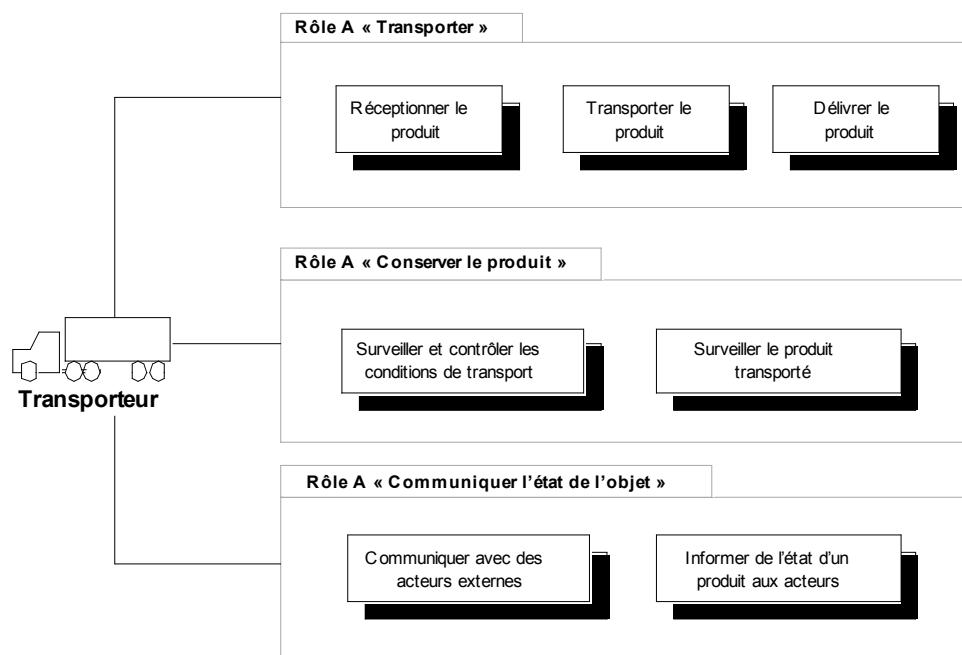


Figure 72 : Moyen de transport des objets communicants et ses rôles.

En résumé, les informations contextuelles des objets communicants déterminent les types de services accessibles pour chaque acteur de la chaîne logistique.

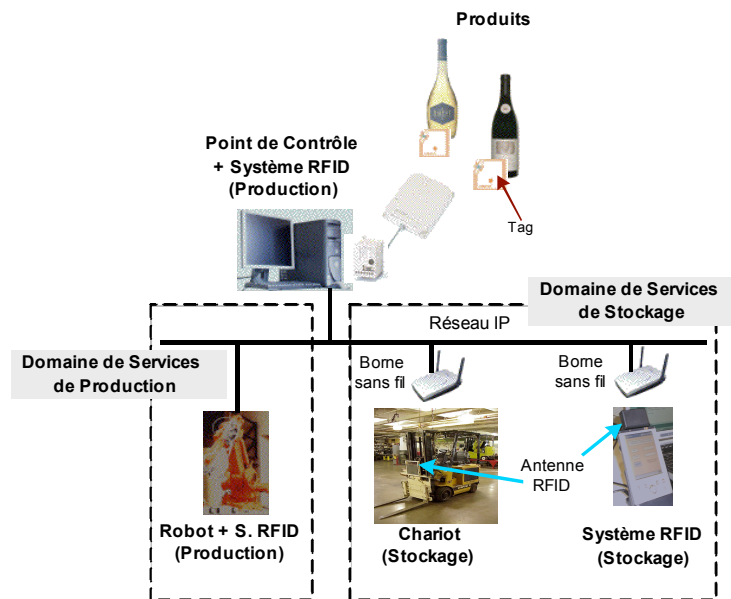


Figure 73 : Dépendances des services d'un objet communicant.

La Figure 73 illustre la relation entre les facteurs déterminant le contexte d'un objet communicant. On observe deux types de produits physiques (instances d'une famille de produits) qui sont équipés avec des étiquettes électroniques. Ces étiquettes stockent les caractéristiques intrinsèques de ces produits (forme, taille, poids, couleur, date de fabrication, ...). Les produits physiques sont identifiés par un dispositif (cp) UPnP avec un système RFID intégré. Ensuite, les services offerts par les produits sont diffusés aux acteurs de la chaîne logistique (point de contrôles UPnP) qui font partie du réseau ambiant UPnP. On observe deux domaines d'interaction pour objets communicants. D'une part, le robot appartient au domaine de services fonctionnel de production, d'autre part, le chariot et le PDA font partie du même domaine de services fonctionnel de stockage et pouvant accéder, à distance, à des services adaptés au profil de chaque utilisateur. Les interactions entre acteurs seront plus détaillées dans les points suivants. Ici, notre volonté est de montrer la relation entre les services disponibles et le contexte d'un objet communicant.

4.3.2 Définition des services d'un objet communicant

Dans ce point, nous allons détailler les éléments dont il faut prendre en compte pour définir les services d'un objet communicant dans notre proposition méthodologique. Les services sont exprimés par des variables d'état et des actions. La Figure 74 représente un diagramme de classes UML décrivant la structure statique d'un objet communicant en incorporant les relations entre le dispositif UPnP (*device*) avec communication RFID, ses Services, ses Variables d'état et ses Actions. Nous avons élaboré ce diagramme sur la base des formalisations proposées par [Jeronimo et Weast, 2003].

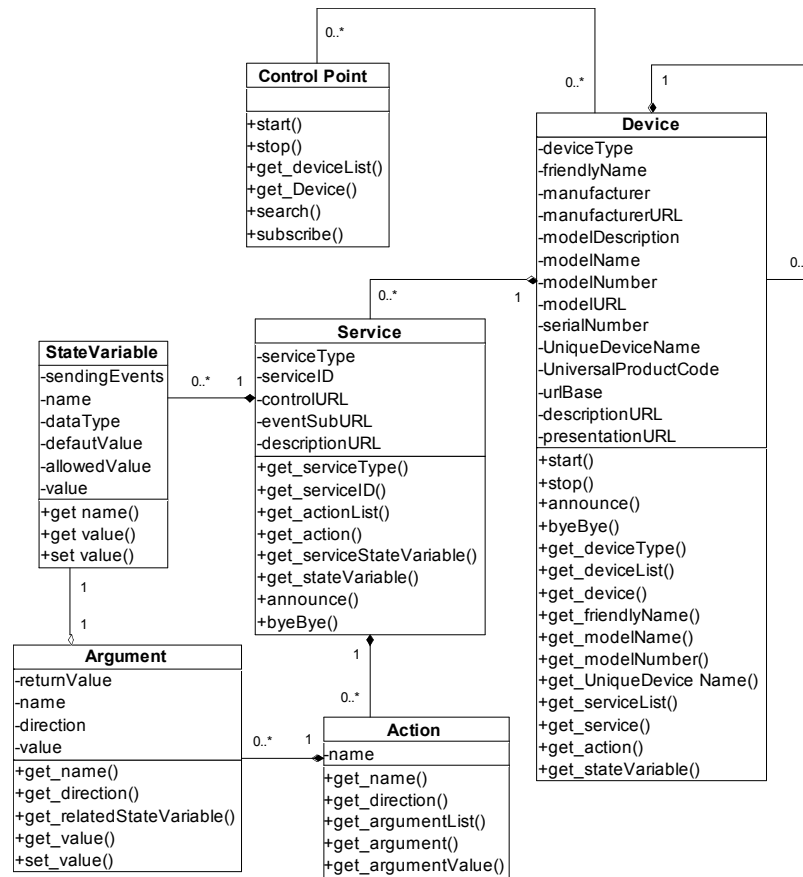


Figure 74 : Diagramme de classes UML décrivant statiquement un objet communicant.

Notre diagramme montre les attributs de chaque classe et les opérations fondamentales associées à chacune d'elles. L'objet communicant géré par le dispositif UPnP est en communication avec aucun ou plusieurs points de contrôles (*control points*). Chaque service a un « service type : URI (*Uniform Resource Identifier*) » pour identifier d'une manière unique le service dans le réseau ambiant. Aussi, chaque service a un « Service ID : URI » pour identifier un service dans les autres services de l'objet communicant. En plus, chaque service a trois URLs apportant l'information nécessaire aux points de contrôle pour communiquer avec ses services. L'URL de contrôle (*ControlURL*) est utilisée pour contrôler les services à distance. L'URL de souscription (*EventSubURL*) est utilisée pour souscrire aux événements générés par l'objet communicant et l'URL de description (*DescriptionURL*) informe les points de contrôle de l'endroit permettant d'obtenir la description détaillée des services de l'objet communicant. Chaque service a zéro ou plusieurs variables d'états associées. Chaque variable d'état a associé un nom (*name*), un type (*type*) et une valeur actuelle (*current value*). Une variable d'état peut avoir aussi un ensemble de valeurs employées (*allowed values*) pour décrire l'intervalle permissible de valeurs pour la variable. Une variable d'état est définie comme événementielle (*sendingEvents*) si un changement dans son état doit être informé aux points de contrôles intéressés. Un service a zéro ou plusieurs actions (*actions*) à exécuter. Chaque action a un nom (*name*). Une action est associée à une liste d'arguments afin d'indiquer le type d'action à exécuter. Chaque argument est en relation avec une variable d'état.

Il faut remarquer que tant les caractéristiques d'un produit physique et les informations stockées sur l'étiquette électronique sont représentées dans le diagramme de classe au moyen des variables d'états appartenant à des services associés à un produit physique. Il est important de signaler qu'un produit physique peut apparaître dans un réseau ambiant en utilisant un dispositif quelconque compatible avec les fonctionnalités UPnP et RFID.

Nous proposons d'utiliser le Tableau 6 pour lister tous les services d'un objet communicant, ses objectifs et la quantité de mémoire nécessaire, exprimée en octets, pour stocker l'ensemble de variables significatives associées aux services dans une étiquette électronique. Ainsi, la mémoire requise pour configurer chaque service dans une étiquette électronique sera déterminée par la somme des octets requis pour les variables associées au service en question.

Services	Objectif du Service	Octets requis sur le tag
Service1	Objectif du Service1	X
Service2	Objectif du Service2	Y
...	...	

Tableau 6 : Description des services d'un objet communicant.

Dans les points suivants nous détaillerons comment nous devons définir les services d'un objet communicant, et en conséquent, comment nous devons définir ses variables d'état et ses actions associées.

4.3.2.1 Définition des variables d'état

Les variables d'état permettent d'opérationnaliser les services. Pour cela, nous devons définir et détailler toutes les variables qui permettent de caractériser chaque service défini dans le Tableau 7. Dans le Tableau 2 nous détaillons les attributs qui doivent être considérés. Ce dernier Tableau montre trois variables associées au Service « ALPHA », avec sa description, son type (entier, string, ...), ses valeurs par défaut, ses valeurs permises, les octets requis pour stocker la variable dans une étiquette électronique et l'unité de mesure des variables (gr, Kg,). Le sens de la dernière colonne « événementiel » sera expliqué dans le point suivant.

SERVICE « ALPHA »							
Variable	Description	Type	V. par défaut	V. permmissibles	Bits sur le tag	U. de mesure	Événementiel
Var1	Desc.Var1	Entier	5	[1, 10]	8	[Kg]	Oui
Var2	Desc.Var2	Entier	60	[50, 100]	8	[g]	Oui
Var3	Desc.Var3	String	Bas	[Bas, Elevé]	16	-	Non
...	...						

Tableau 7 : Exemple de description de variables d'état pour un service « ALPHA ».

4.3.2.2 Définition des événements

L'existence d'événements associés aux variables d'état est signalée dans le Tableau 2 dans la colonne « événementiel ». Si la variable est définie comme événementiel (Oui), les points de contrôle pourront souscrire aux événements du service pour être informés d'un éventuel changement dans sa valeur. Les concepteurs de l'objet communicant doivent définir quelles variables sont du type événementiel.

4.3.2.3 Définition des actions

Dans le Tableau 8 nous montrons les trois types d'actions qui peuvent être déployées par un objet communicant. Une action de type « Get » permet de consulter la valeur d'une variable. Cette action

nécessite une direction de type « output » vers le point de contrôle. Une action de type « Set » permet de changer la valeur d'une variable. Dans ce cas, la direction de l'action est de type « input », c'est-à-dire, vers l'objet communicant. Et finalement, une action de type « To do » permet d'indiquer à l'objet communicant de réaliser une action déterminée de sa propre initiative ou par l'initiative de quelqu'un d'autre (acteur de la chaîne logistique). Dans ce cas, une variable « Résultat » permet de spécifier le résultat de l'action développée.

SERVICE « ALPHA »			
Action	Description de l'action	Variables	Orientation
Get_Valeur_Variable1	Obtenir la valeur actuelle de la Variable1	Variable1	output
Set_Valeur_Variable4	Actualiser la valeur de la Variable4	Variable4	input
Comparer_ValeursV1V2	Comparer la valeur de la Variable1 avec la valeur de la Variable2	Variable1	output
		Variable2	output
		Résultat	output
...	...		

Tableau 8 : Exemple de description de actions à exécuter pour un service déterminé « ALPHA ».

4.3.3 Documents XML décrivant l'objet communicant

La structure d'un objet communicant logique est codée dans un document XML. Ce document est divisé en deux parties. La première partie contient la description du dispositif UPnP - RFID, comme par exemple : le nom du dispositif, le fabricant du dispositif, La deuxième partie du document XML contient une liste de tous les services qui sont supportés par le produit physique et, par conséquent, intégrés dans l'objet communicant. Le document fournit pour chaque service le type de service (*serviceType*), un identifiant (*ServiceId*), une URL pour obtenir la description détaillée du service, une URL pour contrôler l'objet communicant (*SCPDURL*), une URL pour contrôler l'objet communicant (*controlURL*), une URL pour souscrire aux événements (*eventSubURL*) et une URL optionnelle pour accéder à une page web de présentation de l'objet communicant.

La Figure 75 illustre un document XML indiquant le service qui permet identifier un objet au moyen d'un code unique : *ObjectIdentification* (*serviceType*). Dans le document on observe, à titre d'exemple, les URLs de description, de contrôle, d'événements et de présentation afin de pouvoir utiliser ce service.

```

<device>
...
description du dispositif UPnP - RFID
<serviceList>
  <service>
    <serviceType> ObjectIdentification </serviceType>
    <serviceId> ServiceId </serviceId>
    <SCPDURL> http://193.50.39.120:8080/objet/description.xml </SCPDURL>
    <controlURL> http://193.50.39.120:8080/objet/control </controlURL>
    <eventSubURL> http://193.50.39.120:8080/objet/event </eventSubURL>
  </service>
  déclaration d'autres services ici
</serviceList>
<presentationURL>http://193.50.39.120:8080/presentation.html</presentationURL>
</device>

```

Figure 75 : Document XML décrivant globalement un objet communicant

Afin de décrire en détail un service associé à l'objet communicant, on utilise un document XML. Celui-ci contient la liste des actions (*actionList*) supportées par l'objet communicant, ainsi quela description des variables associées aux actions. Par exemple dans la Figure 76 on illustre la structure d'un document XML décrivant le service *ObjectIdentification*, présenté auparavant, en indiquant l'action *GetObjectId* afin d'obtenir l'identification (code) d'un objet. La variable associée à cette action est *ObjectId*. Cette variable, de type *string*, est en « lecture », c'est-à-dire, qu'elle a une direction de type « out ». Cette variable est non événementielle parce que l'identification de l'objet est de nature statique durant son cycle de vie.

```

<scpd>
....
<actionList>
  <action>
    <name> GetObjectId </name>
    <argumentList>
      <argument>
        <name> ArgumentObjectId </name>
        <direction>out</direction>
        <relatedStateVariable> ObjectId </relatedStateVariable>
      </argument>
    </argumentList>
  </action>
</actionList>
<serviceStateTable>
  <stateVariable sendEvents="no">
    <name> ObjectID </name>
    <dataType> string </dataType>
    others state variables here
  </stateVariable>
</serviceStatetable>
</scpd>

```

Figure 76 : Document XML décrivant un service d'un objet communicant.

4.3.4 Classes de services pour un objet communicant

Les services associés à un objet peuvent être dépendants de son information contextuelle représentée, dans notre méthodologie, par la nature de l'objet physique (typologie d'objets), par le domaine d'interaction de l'objet et par le profil de l'acteur de la chaîne logistique. Nous pouvons approcher une

classification des services d'un objet en groupant les services en fonction des phases de son cycle de vie. Dans cette conceptualisation, nous pouvons distinguer autant des classes de service que de phases du cycle de vie appartenant à un objet. Cette approche a été adoptée par [Wong et al., 2002] afin d'explorer les possibles applications de l'approche objet pointeur vers un système d'information (approche **Electronic Product Code**). Une autre approche de classification de services est basée sur le domaine géographique dans lequel l'objet est situé en prenant en compte les droits d'accès à ces services en fonction du profil de l'acteur [Loke et al., 2005].

Nous proposons de classer les services d'un objet communicant en considérant le caractère statique ou dynamique des services dans le cycle de vie de l'objet. Les services d'un objet sont dynamiques quand ils sont dépendants des phases de son cycle de vie et de ses informations contextuelles (type d'objet, domaine d'interaction de l'objet (notion d'espace géographique, état de l'objet et temps), et profil de l'acteur de la phase du cycle de vie). Dans ce cas, les services dynamiques sont en relation directe avec les phases du cycle de vie selon les classes de services suivantes : classe de service production, classe de service transport et distribution, classe de service entreposage, classe de service vente, classe de service usage, classe de service maintenance, classe de service recyclage. D'un autre point de vue, les services d'un objet sont statiques quand ils sont indépendants de facteurs externes et sont disponibles d'une façon permanente durant son cycle de vie. Un acteur de la chaîne logistique peut avoir une vision partielle ou globale de ces services en fonction de son profil. Nous identifions trois classes de services statiques : classe de services identification de l'objet, classe de service information sensitive de l'objet et classe de service traçabilité de l'objet. La Figure 77 illustre dans la partie inférieure les services dynamiques et ses classes de services associés au cycle de vie de l'objet, et dans la partie supérieure les services statiques et ses classes de services : identification, information sensitive et traçabilité. On observe dans la figure que chaque classe de service contient un ensemble de fichiers en format XML décrivant les services conformant la classe.

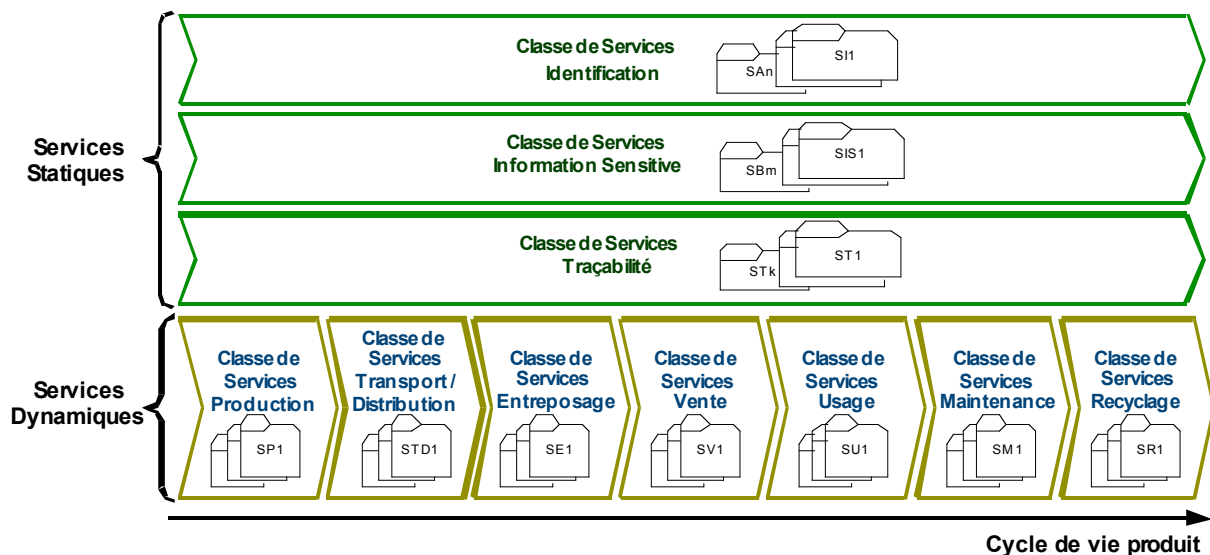


Figure 77 : Déroulement du Cycle de vie et Classes de Services associées des Produits Communicants.

- **Classe de services d'identification d'un objet physique :**

Le but de cette classe de service est de permettre d'identifier un objet physique de manière unique durant son cycle de vie. Par exemple nous pouvons utiliser le code EPC (**Electronic Product Code**) pour identifier le code de l'entreprise fabricant de l'objet, le code du type de l'objet et le code sérialisé de l'objet (Voir le point 2.2.1.1 pour obtenir plus d'informations sur l'initiative EPC Global). La technologie d'identification automatique RFID permet de stocker un code d'identification unique dans

une étiquette électronique attachée à un objet et permettant la communication entre l'objet / produit physique et sa représentation virtuelle. Optionnellement, des attributs supplémentaires d'identification d'objets peuvent être stockés localement dans l'objet, comme par exemple : le modèle de l'objet, la forme particulière de l'objet, ... Dans une vue plus large, d'autres technologies d'identification automatique d'objets peuvent être employées : code barre, signature biométrique de produits [Fuentelba, 2005], signature informatique, ...

- **Classe de service de gestion de l'information sensitive de l'objet :**

L'objectif de cette classe de service est de grouper les informations qui peuvent être collectées par l'objet au moyen de capteurs intégrés dans l'objet physique. Par exemple les variables qu'on peut mesurer en utilisant des capteurs sont : la température, l'humidité, le son, la lumière, le mouvement, la proximité, la force / pression [Beigl et al., 2004].

Un capteur de température permet de déterminer la température (degrés Celsius) de l'environnement associé à l'objet (par exemple durant son transport, son stockage, son usage, ...), ou plus simplement, permettre de distinguer entre deux ou plusieurs états associés à l'objet : niveau de température correcte ou incorrecte, ... Un capteur d'humidité permet de mesurer le niveau d'humidité de l'air afin de surveiller l'environnement proche d'un objet sensible à ces valeurs pendant son transport, stockage, vente, usage, ... Un capteur de sons permet de clarifier le contexte d'un produit en déterminant le niveau d'activité dans une zone et en discriminant différentes situations associées à l'objet (niveau de bruit durant le transport, stockage, etc.). Un capteur de lumière peut être utilisé pour inférer la localisation d'un objet dans son environnement sur la base de patrons spectraux de lumière. Un capteur de mouvement permet de déterminer l'état d'un objet (déplacement) et grâce à cela, il est possible de distinguer différents types d'activités développées par l'objet, ou plus généralement, il est possible de déterminer la situation associée à l'objet. Un capteur de proximité peut être utilisé pour activer un processus et pour inférer le niveau d'activité autour de l'objet dans un endroit déterminé. Un capteur de force ou pression permet d'interagir avec l'objet afin de lui demander une action (fonctionnalité d'interface utilisateur). Ou bien, avec ce type de capteur, il est possible d'inférer l'état d'un objet au moyen des changements de la distribution de son poids dans le temps (lorsque l'objet est conteneur d'autres objets, de liquide, ..).

La considération simultanée des différentes variables sensibles permet de préciser le contexte d'un objet et ainsi il est possible d'activer des services contextuels qui seront exécutés par l'objet lui-même, ou seront offerts aux acteurs de la chaîne logistique. On constate que les mesures des variables des capteurs représentent des services statiques mais les services contextuels qui émergent de ces mesures sont de type dynamique et sont en relation avec les phases du cycle de vie de l'objet.

- **Classe de service de traçabilité des objets :**

Le concept de traçabilité d'objet est associé à la capacité de suivre la trajectoire d'un objet (localisation ou *tracking*) au travers de la chaîne logistique et à la capacité de retrouver l'origine d'un objet dans la chaîne logistique au moyen des informations enregistrées (*tracing*) [Jansen-Vullers et al., 2003]. Selon la norme [ISO 8402, 1994] la traçabilité est l'aptitude à retrouver l'historique, l'utilisation ou la localisation d'une entité au moyen d'identifications enregistrées. Le dictionnaire APICS définit la traçabilité (dans le domaine logistique) comme : 1) l'attribut qui permet de déterminer la localisation continue d'une expédition, 2) la sauvegarde et la localisation par lot ou numéro de série des matières premières, des composants et des processus utilisés dans la production [APICS, 1998].

Dans notre cas, la classe de service traçabilité a pour but de permettre de localiser individuellement un objet, durant son cycle de vie, dans un domaine d'interaction donné (espace géographique, forme/état

d'un objet, temps) et de retrouver les événements, les changements d'état et les transformations effectuées sur l'objet par des acteurs de la chaîne logistique (processus, utilisateurs ou autres objets). La traçabilité d'un objet est possible grâce à l'intégration entre les technologies d'identification automatique et les technologies de systèmes ambiants.

- **Classes de services dynamiques dans le cycle de vie d'un objet :**

Dans cette catégorie de services, nous distinguons les classes de services d'un objet qui évoluent en fonction des phases du cycle de vie de l'objet, et plus précisément, en fonction du type d'objet, du domaine d'interaction de l'objet (espace géographique, forme/état de l'objet et temps) et des acteurs de la chaîne logistique. Nous utilisons les phases du cycle de vie comme critère de classification représentant un domaine de services. Ainsi, une phase du cycle de vie est vue comme un domaine de services fonctionnel pouvant avoir un ou plusieurs domaines de services géographiques associés. La vision partielle ou complète du domaine fonctionnel sera déterminée par le profil de l'acteur de la chaîne logistique. Le domaine d'interaction de l'objet est associé à la présence de l'objet (au travers de ses formes / états) dans un domaine géographique dans un instant donné. Si l'objet change d'état, produit d'une transformation déclenchée par un événement, on peut dire que l'objet change de domaine d'interaction. La Figure 78 illustre un domaine fonctionnel associé à une phase du cycle de vie d'un objet (phase A). On observe que ce domaine fonctionnel est composé de trois domaines géographiques (A,B et C). Aussi on peut observer que dans chaque domaine géographique, il y a des domaines d'interaction (A1, B1, B2 et C1) qui sont associés à la présence d'objets du type X dans le domaine géographique correspondant. Chaque domaine d'interaction est créé pour un objet de type X dans un espace géographique déterminé à un moment donné. Par exemple, dans le domaine géographique B, un objet de type X peut changer d'un domaine d'interaction B1 à un domaine d'interaction B2 parce qu'un événement a déclenché une transformation sur l'objet. Ensuite, la figure montre que l'objet arrive au domaine géographique C en activant un domaine d'interaction C1. L'acteur de la chaîne logistique, indiqué dans la figure, aura une vision du domaine fonctionnel en accord avec ses droits d'accès.

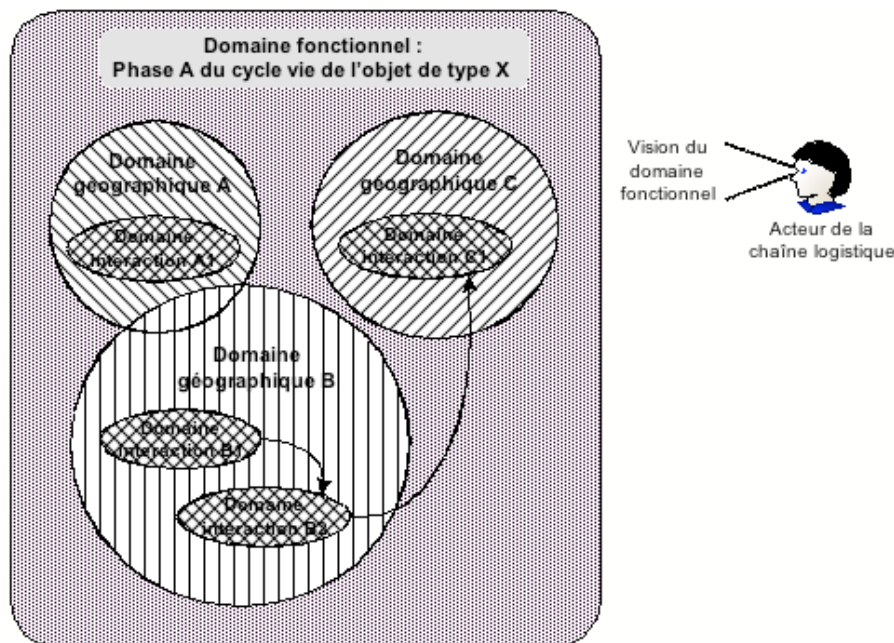


Figure 78 : Domaines géographiques dans un domaine fonctionnel d'un objet communicant

Comme exemple nous citons la phase d'entreposage des objets physiques. Dans ce cas, le domaine fonctionnel est composé par un ensemble de domaines géographiques : des endroits de stockage

d'objets. On peut imaginer que chaque endroit offre des caractéristiques d'entreposage adaptées au type d'objet.

Comme type d'objet nous pouvons considérer : des produits physiques de différents types (produits surgelés, produits laitiers, produits en boîte de conserve, etc., des types de boîtes (cubique, ...) regroupant des produits physiques et des types de palettes (homogènes, hétérogènes, ...) regroupant des boîtes de produits. Comme acteurs de la chaîne logistique nous pouvons considérer : des points d'entrée, des points de sortie, des chariots de transport, des utilisateurs humains, des objets,

Pour ce cas, il y aura un ensemble de services conformant une classe de service entreposage d'objets. Comme exemple de service nous considérons les « conditions de stockage des objets » contenant des services pour faire interagir les objets avec les acteurs de la chaîne logistique afin de gérer les conditions de stockage des objets, depuis leur entrée dans l'entrepôt (domaine fonctionnel), en passant par leur stockage dans des endroits adaptés (domaines géographiques) aux types d'objets, jusqu'à la sortie des objets en considérant les exigences des objets physiques pour être stockés correctement pendant le temps nécessaire. Des services peuvent être activés sur l'objet si une condition est non respectée. Ceci implique que l'objet effectue une demande de traitement immédiate vers les acteurs de la chaîne logistique concernés. Dans le Chapitre 5 nous analyserons plus en détail l'entreposage des objets communicants.

4.4 Modélisation des interactions entre objets communicants et acteurs de la chaîne logistique

Selon notre approche, lorsque le produit physique est en communication avec le dispositif UPnP équipé de capacités de traitement RFID, le produit physique est transformé automatiquement en un objet communicant ou produit intelligent pouvant offrir ou demander des services dans un réseau ambiant. En ce sens, dans cette partie du chapitre nous détaillerons les interactions de base entre un objet communicant et des acteurs de la chaîne logistique, représentés par des points de contrôle et des dispositifs UPnP dans un réseau ambiant.

Les interactions à analyser sont : l'annonce d'un objet communicant dans un réseau ambiant, la découverte d'un objet communicant et de ses services, la description des services d'un objet communicant par un dispositif (cp) UPnP, l'invocation des services d'un dispositif UPnP par un objet communicant actif, l'invocation des services d'un objet communicant par un objet communicant actif et la souscription aux événements d'un objet communicant par des dispositifs (cp) UPnP ou des objets communicants actifs. Nous utilisons le langage de modélisation UML [OMG UML, 2004] [Fowler, 2004] [Bauer et Odell, 2005] [Willard, 2006] pour représenter les interactions entre les objets et les acteurs de la chaîne logistique. En particulier, le diagramme de séquence sera employé pour représenter les messages entre les entités.

4.4.1 Annonce d'un objet communicant et de ses services dans un réseau ambiant

L'envoi d'annonces est un mécanisme qui permet de découvrir les services [Zhu et *al.*, 2002] dans un réseau ambiant UPnP. Dans notre proposition méthodologique, lorsqu'un produit physique arrive dans une zone d'attraction RFID gérée par un dispositif UPnP avec un système RFID intégré, le dispositif UPnP - RFID envoie automatiquement des messages d'annonces vers les points de contrôle présents dans le réseau ambiant. Le diagramme de séquence UML présenté dans la Figure 79 illustre les interactions entre un produit physique, sa contrepartie logicielle (Dispositif UPnP - RFID), et un Point de contrôle (cas 1 de notre méthodologie). On peut observer que quand le produit arrive à la zone d'attraction RFID automatiquement le Point de contrôle est informé de ce fait. Cette action permet d'initier un éventuel processus d'appel de services si le point de contrôle est intéressé par les services offerts par l'objet communicant. Lorsque le produit physique abandonne le champ d'attraction RFID, les points de contrôle sont informés de ce fait automatiquement.

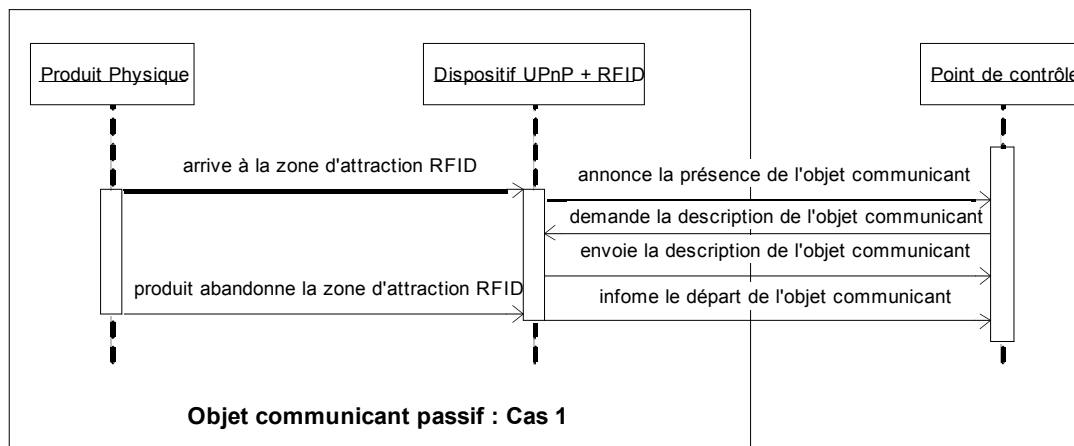


Figure 79 : Diagramme de séquence UML : Annonce d'un objet communicant passif sur le réseau UPnP.

Les trois cas proposés par notre méthodologie sont concernés par ce mécanisme de découverte. Grâce à cela, un produit physique devient un objet communicant en signalant automatiquement sa présence à tous ceux qui pourraient avoir intérêt d'interagir avec lui.

4.4.2 Découverte d'un objet communicant et de ses services

Lorsqu'un point de contrôle arrive au réseau UPnP après l'arrivée de l'objet communicant, le point de contrôle peut faire une recherche sur le réseau pour s'informer s'il y a des objets communicants qui l'intéressent. Cette deuxième méthode de découverte de services est présentée dans la Figure 80 montrant le diagramme de séquence UML du processus de découverte d'un objet communicant passif et de ses services par un point de contrôle UPnP. Cette Figure montre la recherche effectuée par le Point de Contrôle B afin d'évaluer les possibles interactions futures avec les objets communicants reliés au réseau UPnP. Ensuite, l'objet communicant passif lui envoie sa description logique et la description de tous ses services au moyen des fichiers XML. Ce type d'interaction est caractérisé par une recherche réalisée par l'initiative des points de contrôle.

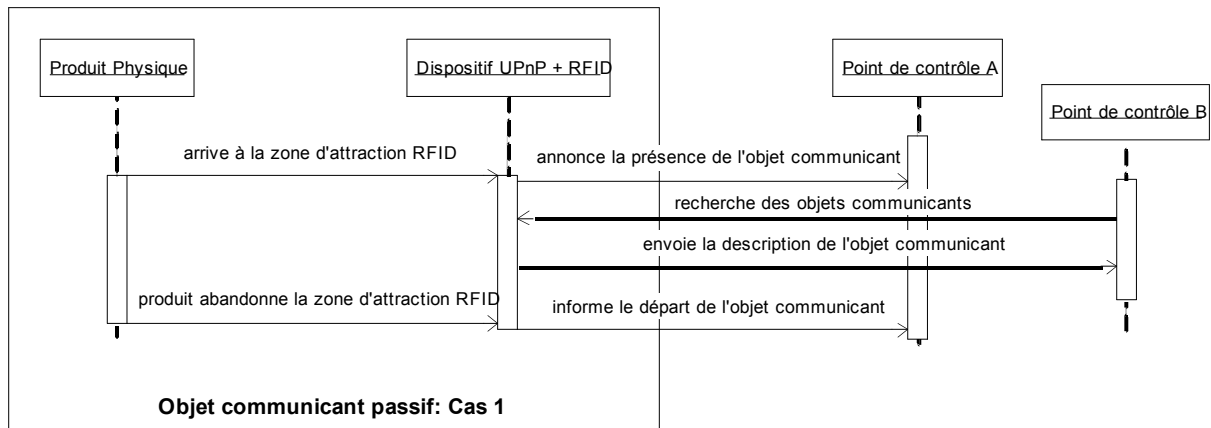


Figure 80 : Diagramme de séquence UML : Découverte d'un objet communicant passif par un point de contrôle UPnP.

Comme exemple de points de contrôle, dans la phase de production, nous pouvons citer : des machines, des robots, des chariots de transport, des utilisateurs humains étant seulement des demandeurs de services. Ces demandeurs de services peuvent interagir avec l'objet communicant à distance.

Par rapport à notre méthodologie, les cas concernés par ce mécanisme de découverte sont le deuxième, objet communicant actif, et le troisième, objet communicant actif unifié. Ces deux cas, permettent de caractériser un produit physique comme un objet communicant avec des capacités d'invoquer des actions vers des dispositifs UPnP. Par exemple dans la Figure 81, le Dispositif (cp) UPnP représente un produit physique intégré dans l'objet communicant (cas 3 de notre méthodologie) qui recherche des services dans un réseau UPnP. Dans ce dernier cas, il s'agit d'une communication produit intelligent - produit intelligent. Ces interactions du type objet – objet seront expliquées dans les points suivants.

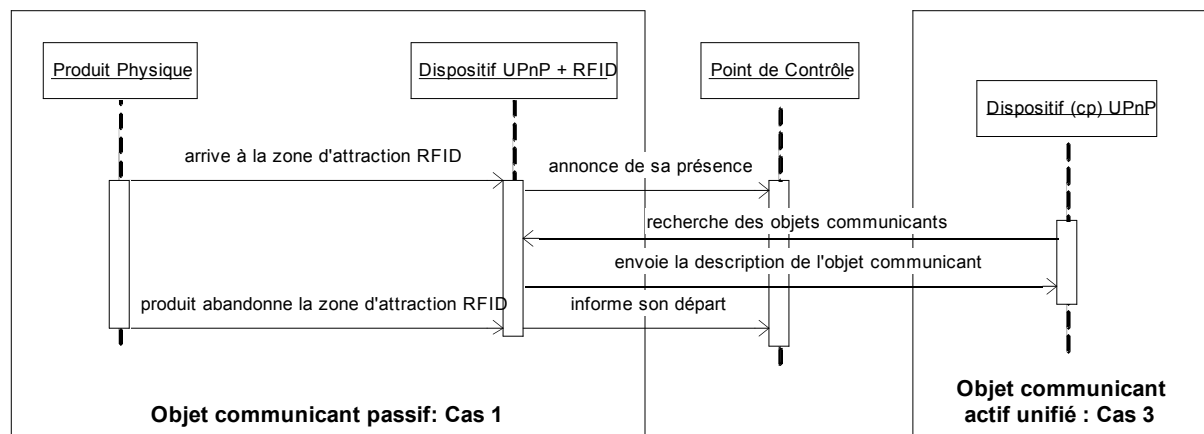


Figure 81 : Diagramme de séquence UML : Découverte d'un objet communicant passif par un objet communicant actif.

4.4.3 Description des services d'un objet communicant

Dans le cas où un point de contrôle est intéressé par les services offerts par l'objet communicant, le point de contrôle envoie une demande de description de services plus détaillée afin d'obtenir plus d'information sur chaque service supporté. Dans la Figure 82 nous pouvons observer la demande de description de services réalisés par le point de contrôle à l'objet communicant. Ensuite, l'objet

communicant passif envoie au point de contrôle la description de tous ses services en indiquant les actions supportées, ses variables d'état et son caractère événementiel s'il correspond.

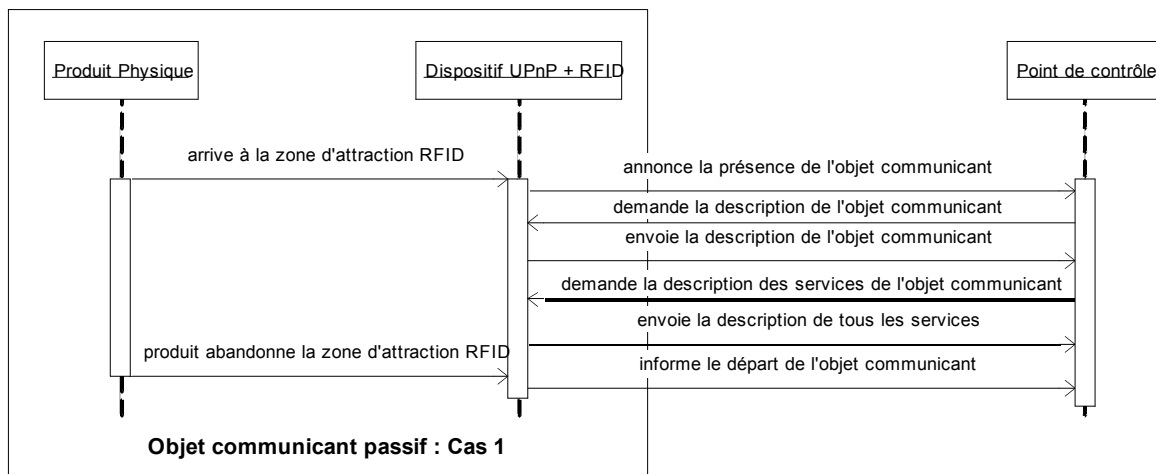


Figure 82 : Diagramme de séquence UML : Description des services d'un objet communicant passif.

Les objets communicants actifs sont ceux qui demanderont la description de services des objets communicants dans un réseau UPnP. Selon notre proposition, les objets communicants qui peuvent demander des services sont les produits physiques enrichis avec leur image informationnelle grâce à des dispositifs UPnP équipés de technologie de communication RFID (cas 2), les produits physiques intégrés dans des dispositifs (cp) UPnP (cas 3) et les points de contrôle UPnP.

4.4.4 Invocation des services d'un objet communicant par un dispositif

Après qu'un demandeur de services a récupéré la description des services supportés par un objet communicant, le demandeur de services invoque une action à distance sur ce dernier. Le Diagramme de séquence UML de la Figure 83 illustre les interactions entre un Dispositif (cp) UPnP (cas 3 de la méthodologie) et un objet communicant passif (cas 1 de la méthodologie). Cette figure illustre les trois types actions qu'un demandeur de service peut réaliser : obtenir la valeur d'une variable, changer la valeur d'une variable et demander d'exécuter une action spécifique « to do » à l'objet communicant. D'abord le Dispositif (cp) UPnP demande une action du type « Get » pour obtenir la valeur de la variable Var1, soit sur l'étiquette électronique soit sur une base de données. Ensuite avec la réponse, il décide de demander une action du type « Set » pour changer la valeur de la variable Var1. Finalement, le Dispositif (cp) UPnP invoque une action « to do » afin de réaliser une action déterminée sur l'objet communicant. Tout de suite il reçoit le résultat de l'action.

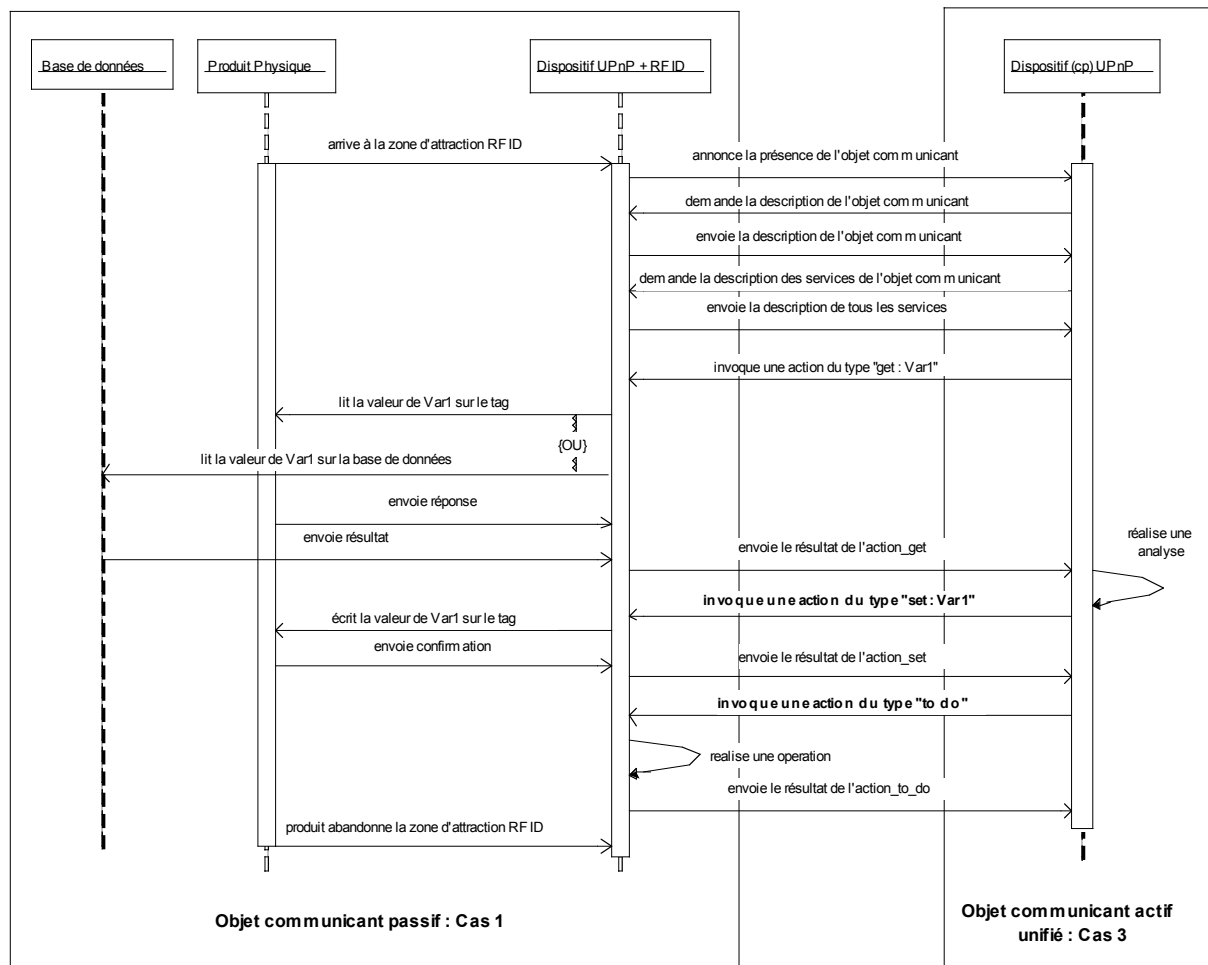


Figure 83 : Diagramme de séquence UML : Invocation d'action sur un objet communicant passif.

4.4.5 Invocation des services d'un dispositif par un objet communicant actif

Dans ce cas, l'objet communicant actif réalise une recherche des dispositifs UPnP dans le réseau afin de trouver un dispositif adapté à ses besoins. Si l'objet communicant actif trouve le dispositif adéquat, il pourra envoyer des messages d'invocation d'actions vers le dispositif. L'invocation de ces actions lui permet de disposer de l'information nécessaire pour prendre des décisions. Le diagramme de séquence UML présenté dans la Figure 84 illustre les interactions entre un objet communicant actif (cas deux de notre méthodologie) et un dispositif UPnP relié au réseau UPnP. Après avoir trouvé le dispositif UPnP recherché, l'objet communicant actif interroge le dispositif UPnP afin de connaître la description de ses services et la liste de ses actions. Ensuite, l'objet actif est en condition de sélectionner les actions les plus appropriées selon son état actuel et de les exécuter. Ce processus peut se répéter autant de fois qu'il est nécessaire afin de prendre les décisions correctes au moment correct.

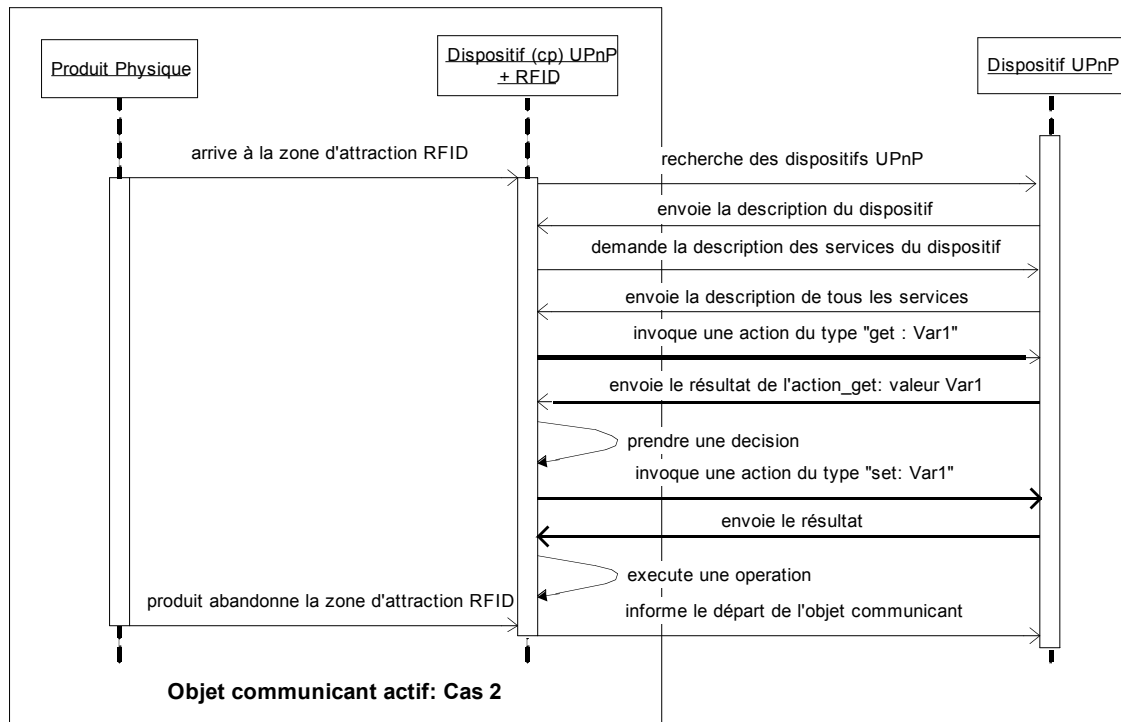


Figure 84 : Diagramme de séquence UML : Invocation d'action réalisée par un objet communicant actif.

4.4.6 Invocation de services d'un objet communicant par un objet communicant actif

Dans ce cas, l'objet communicant actif peut invoquer des actions vers d'autres objets communicants passifs ou actifs. En d'autres termes, il s'agit d'une communication produit intelligent - produit intelligent. En termes pratiques, avec cette configuration méthodologique il est possible de faire communiquer un produit physique avec un autre produit physique en utilisant l'infrastructure de services que nous avons définie. Le diagramme de séquence UML de la Figure 85 illustre une communication objet communicant - objet communicant. Tout les deux sont des objets communicants actifs pouvant offrir ou demander des services. Nous montrons dans la figure l'invocation des actions entre les objets communicants. Dans cette illustration, la lecture d'information sur l'étiquette électronique de chaque produit physique représente le mécanisme d'interaction de base. Dans ce sens, des questions telles que : Qui est-tu ? , Où vas-tu ? , Es-tu disponible maintenant ? ... peuvent être traitées avec ce mécanisme d'interaction.

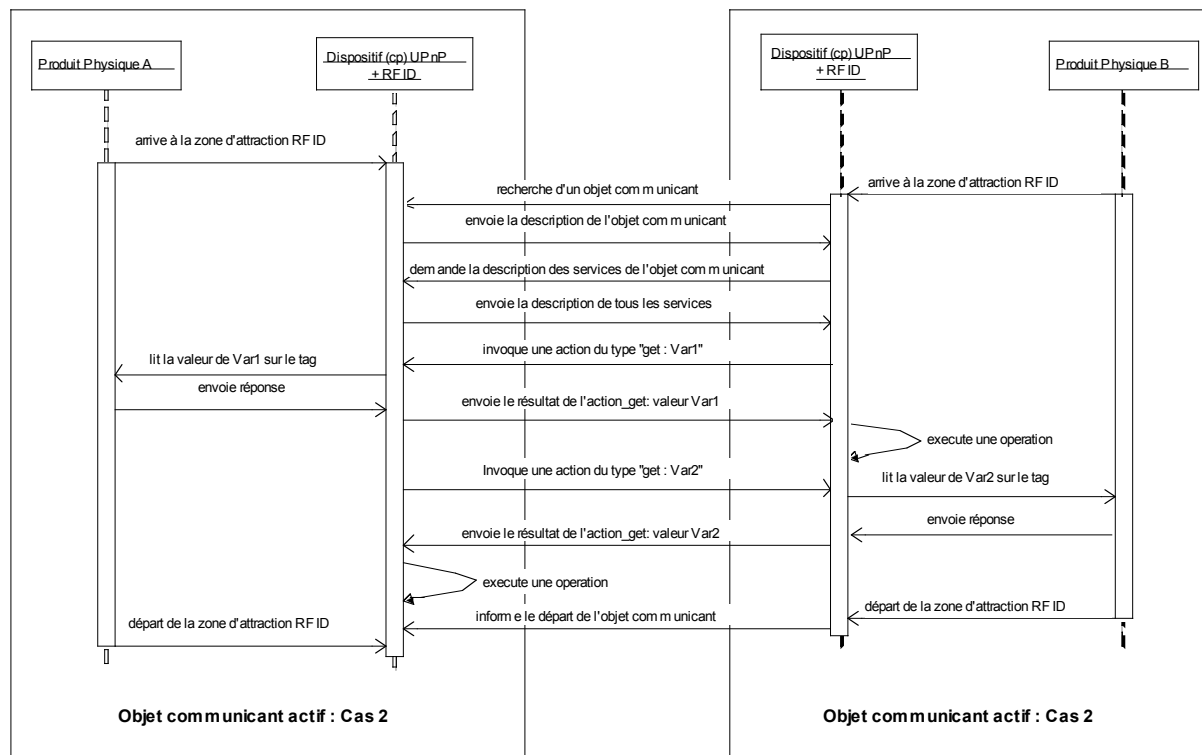


Figure 85 : Diagramme de séquence UML : Invocation d’actions entre objets communicants.

On peut imaginer des interactions plus avancées associées à des services adaptés à des contextes d’études particuliers. Par exemple, les interactions entre produits physiques complémentaires dans leurs fonctionnalités, ou substitution, sont des scénarios potentiellement importants à envisager.

4.4.7 Souscription aux événements d’un objet communicant dans un réseau ambiant

Le mécanisme d’interaction basé sur les services et les événements permet aux points de contrôle d’être informés à tout moment sur les changements des valeurs des variables qui sont critiques pour sa gestion. Dans notre méthodologie, les objets communicants actifs (cas 2 et 3), ainsi que les points de contrôle intéressés, peuvent souscrire aux événements générés par les objets communicants. Le diagramme de séquence UML présenté dans la Figure 86 illustre la requête de souscription, par rapport à la variable Var1, faite par le point de contrôle vers l’objet communicant passif (case 1 de la méthodologie). Après que l’objet communicant effectue un changement dans cette variable, le point de contrôle est informé de cette modification par l’objet communicant.

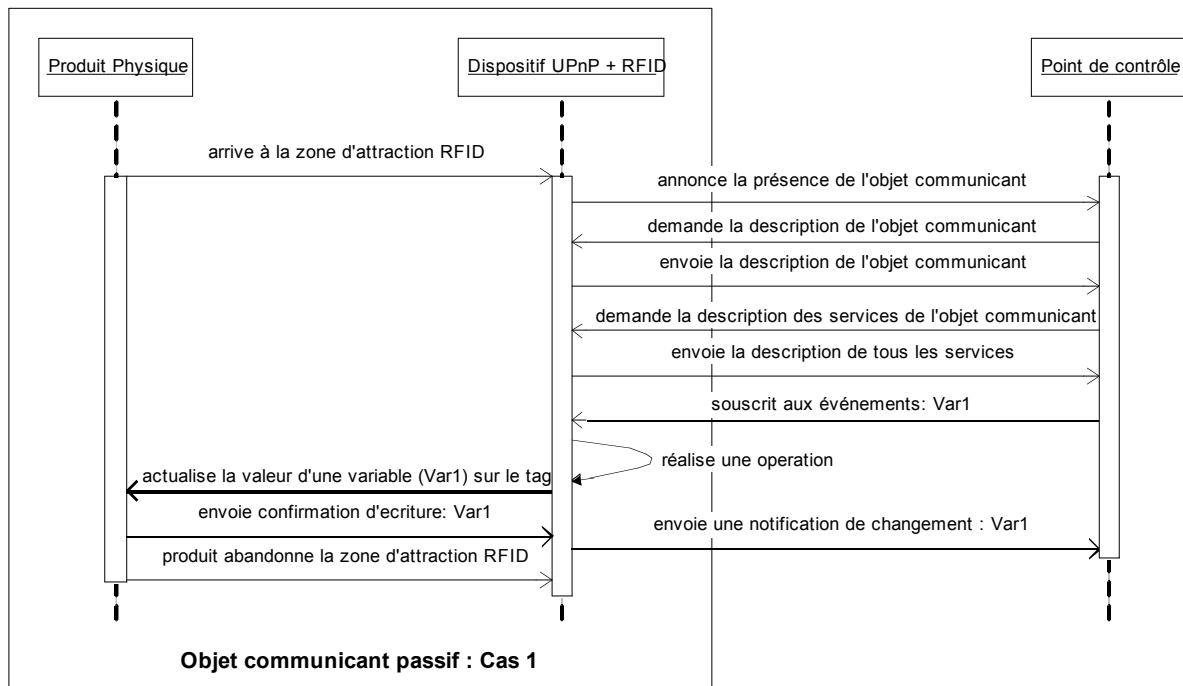


Figure 86 : Diagramme de séquence UML : Souscription aux événements d'un produit communicant.

4.5 Bilan de la méthodologie proposée

Dans ce chapitre, nous avons présenté une proposition méthodologique pour concevoir des objets communicants intégrés dans un réseau ambiant afin d'offrir ses services ambiants dans son cycle de vie. Nous distinguons deux types d'objets communicants : l'objet communicant passif et l'objet communicant actif. L'objet communicant passif a un rôle de fournisseur de services et l'objet communicant actif a un rôle de demandeur de services. Notre proposition méthodologique permet d'identifier un objet communicant automatiquement de manière unique en utilisant la technologie RFID.

Dans notre proposition, les objets communicants obtiennent une adresse logique automatiquement dans un réseau ambiant. La localisation de ces objets est réalisée grâce à des méthodes directes ou indirectes. La découverte des services des objets communicants est effectuée sur la base de mécanismes d'annonce et de recherche de services. La description de services d'un objet communicant est basée sur le langage de description XML. Un objet communicant actif peut effectuer une invocation d'actions à distance. La souscription aux événements générés par un objet communicant peut être réalisée par d'autres objets communicants actifs ou par des points de contrôle dans un réseau ambiant.

Nous avons classé les services en deux types : services statiques et services dynamiques. Les services statiques sont présents durant tout le cycle de vie de l'objet. Les services dynamiques sont adaptés à chaque phase du cycle de vie de l'objet. Les phases du cycle de vie de l'objet représentent ses domaines fonctionnels. Ces domaines fonctionnels sont composés de domaines géographiques. La présence d'un objet est associée à un domaine d'interaction représentant l'objet et son état dans un espace géographique déterminé à un moment donné. Les domaines d'interactions des objets sont activés dans les domaines géographiques. Ainsi, les services associés à un objet communicant sont déterminés par le type d'objet communicant, son domaine d'interaction et le profil de l'acteur de la chaîne logistique.

La modélisation de services permet de spécifier en détail les services par un objet communicant en définissant ses actions et ses variables d'action. Nous avons modélisé les interactions de base entre des objets communicants et des acteurs de la chaîne logistique. La modélisation d'interactions pour des cas d'application spécifiques nécessite une formalisation détaillée en considérant les processus métiers et ses relations avec les fonctionnalités génériques d'un objet communicant. La couche logicielle des objets communicants sera déterminée par les règles d'opération définies par un objet communicant, les processus concernés et les acteurs de la chaîne logistique. La modélisation de services et des interactions des objets communicants est basée sur le langage de modélisation UML.

Capacité de l'objet communicant	Moyen d'obtention	Exemple
Posséder une identification unique	Identification automatique d'un produit physique par radiofréquence.	Produit physique tagué avec une étiquette électronique de 13.56 MHz.
Communiquer avec son environnement	L'objet peut communiquer avec des dispositifs UPnP, points de contrôles UPnP, dispositifs (cp) UPnP et des autres objets communicants.	Un produit physique communique son arrivée et son départ dans un réseau ambiant.
Mémoriser et gérer des informations propres	Le produit physique stocke des informations propres sur son étiquette électronique et sur des bases de données distantes.	Le produit stocke son code d'identification, sa date de fabrication, son poids, ...
Disposer d'un langage de dialogue	L'objet communique grâce aux protocoles UPnP en annonçant sa présence et ses services, et en demandant des services distants.	L'objet communicant utilise les protocoles SOAP et GENA pour se communiquer avec des autres entités dans un réseau UPnP.
Participer aux processus de décision durant son cycle de vie	Le produit intelligent grâce à une couche logicielle intégrée dans sa représentation UPnP peut incorporer des règles de décisions appropriées afin de gérer son comportement.	Dans la couche logicielle d'un objet communicant il est possible d'implémenter des mécanismes d'autoévaluation, autodiagnostic,....
Surveiller et contrôler son environnement	L'objet peut communiquer avec des capteurs UPnP distants afin de recevoir des informations caractéristiques de son environnement.	Dans la couche logicielle il est possible d'implémenter des mécanismes de comparaison des variables essentielles du produit versus des variables externes : température, humidité,
Offrir des services sur son cycle de vie	L'objet communicant passif est un fournisseur de services.	L'objet communicant offre de l'information sur son identification, sa procédure de fabrication, sa procédure de recyclage
Demander de services aux acteurs participant dans son cycle de vie	L'objet communicant actif est un fournisseur de services et un demandeur de services.	L'objet communicant peut demander des services vers d'autres entités UPnP : disponibilité de ressources, ...

Tableau 9 : Synthèse et implémentations des capacités d'un Produit Communicant

La Tableau 9 représente une synthèse des capacités nouvelles conférées aux produits physiques par l'approche Produit Communicant proposée, et de leurs réalisations dans les objectifs de mémorisation, communication, action et décision. La première colonne reprend les capacités attendues d'un objet communicant. La deuxième colonne exprime comment ces capacités sont atteintes par notre proposition méthodologique. Enfin, la troisième colonne exprime un exemple de mise en place réelle de chaque capacité d'un l'objet communicant.

4.6 Conclusions

Nous avons montré que notre proposition méthodologique permet de transformer un objet ou produit physique en un objet communicant ou produit intelligent avec des capacités de mémorisation, de perception, de communication, d'action et de décision avec son environnement. L'objet communicant est représenté par l'ensemble des composants suivants : un produit physique, une étiquette électronique, un dispositif UPnP avec un système RFID intégré.

Les services représentent l'image informationnelle de l'objet communicant. L'objet communicant est identifié automatiquement grâce à son code d'identification stocké sur une étiquette électronique RFID. Les fonctionnalités de gestion de services ambiants sont obtenues en utilisant les mécanismes fournis pour UPnP. Une couche logicielle intégrée dans l'objet communicant permet de lui donner la capacité d'analyser son environnement et de prendre des décisions par rapport à son destin. Les acteurs de la chaîne logistique pourront interagir avec les objets communicants en fonction de leurs profils. Notre proposition méthodologique considère les interactions : objet - acteur, objet - dispositif et objet - objet.

Notre proposition méthodologique a des impacts significatifs sur l'évolution des relations et interactions entre les acteurs de la chaîne logistique industrielle. Nous analyserons cet impact sur les quatre axes suivants : la propriété intellectuelle des services d'un objet communicant, une approche homogène pour définir les services associés avec le cycle de vie de l'objet, la standardisation des services selon le savoir-faire industriel, et l'utilisation des standards IP afin de communiquer les acteurs de la chaîne logistique.

Notre proposition permet d'offrir des services avancés pour des objets communicants en respectant la propriété intellectuelle des concepteurs d'objets communicants. En effet, notre proposition peut être utilisée sans obliger les concepteurs des objets communicants à divulguer leur savoir-faire et propriété intellectuelle. Seulement les capacités étendues ou services des objets communicants seront diffusés. Les objets communicants sont présents dans des différents domaines d'interactions au travers de ses services. La description des services des objets communicants est faite par des documents XML. Ainsi l'interopérabilité entre objets communicants est garantie. Cependant, la couche logicielle est propriété des concepteurs des objets communicants. Pourtant, il est souhaitable de générer un consensus entre des fournisseurs des objets communicants afin de standardiser la couche logicielle des objets communicants et de proposer un mécanisme d'intégration uniforme pour « charger » les fonctionnalités d'un objet communicant dans un dispositif (cp) UPnP quelconque. Ceci permettra d'intégrer facilement les services d'un nouveau type d'objet communicant dans un réseau ambiant dynamique.

Dans notre proposition, nous favorisons une approche homogène pour doter les objets communicants de services dans leurs cycles de vie. L'homogénéité de l'approche sur le cycle de vie d'un objet est basée sur une définition de services standardisés par l'usage de documents XML interchangeables entre des applications informatiques (fixes ou mobiles) via des technologies Internet. Cette approche permet de classer les services selon des domaines d'interactions spécifiques. Notre proposition est ouverte à l'intégration des services novateurs, définis par les producteurs d'objets communicants, afin d'enrichir les potentialités d'interaction entre les acteurs de la chaîne logistique et les objets physiques.

Le savoir-faire industriel est implicite dans la formalisation des services spécifiques des objets communicants pendant leur cycle de vie. Ces services sont modélisés selon un standard ouvert et flexible basé sur le langage XML. La définition d'actions et de variables d'état est de la responsabilité

des concepteurs des objets communicants. Les mécanismes d'action et de décision associés aux objets communicants sont particuliers à chaque cas d'application. Ceci doit être étudié avec les partenaires impliqués dans le cycle de vie de l'objet communicant.

Notre proposition méthodologique est basée sur des standards IP. Ceci implique l'interopérabilité entre des applications, des équipements et des systèmes d'exploitation. Ce point est très important en considérant la diversité des équipements informatiques appelés à communiquer dans le cycle de vie d'un objet communicant.

5 Application de la méthodologie : cas d'étude d'un entrepôt

5.1 Description du cas d'étude

Dans ce chapitre nous allons appliquer notre proposition méthodologique à la gestion des produits et objets physiques dans un entrepôt. Un entrepôt est un lieu de stockage temporaire d'une grande quantité de produits. Les entrepôts peuvent être utilisés par des entreprises manufacturières, entreprises importatrices / exportatrices de produits, entreprise à gros volumes, transporteurs logistiques... Le but commun de ces entreprises est de stocker des produits temporairement avant qu'ils soient destinés à la suivante phase de la chaîne logistique, soit par exemple : le distributeur, le détaillant ou le client final.

Nous avons choisi ce cas d'étude parce que l'entreposage de produits physiques représente une activité commune à tous les secteurs industriels, nous permettant, de cette manière, d'appliquer notre méthodologie, en obtenant un haut impact industriel dans la chaîne logistique. En effet, notre méthodologie permet d'automatiser les interactions entre les produits, les processus et les utilisateurs, afin d'atteindre de hauts niveaux de performance dans l'entreposage des produits (opportunité, efficacité et productivité). Les processus classiques qui sont développés dans un entrepôt sont : la réception des objets physiques, les déplacements des objets physiques à l'intérieur de l'entrepôt, le stockage temporaire d'objets physiques sur de grandes étagères et l'expédition d'objets physiques vers des centres de distribution, des entreprises, des clients... [Metro Group, 2004].

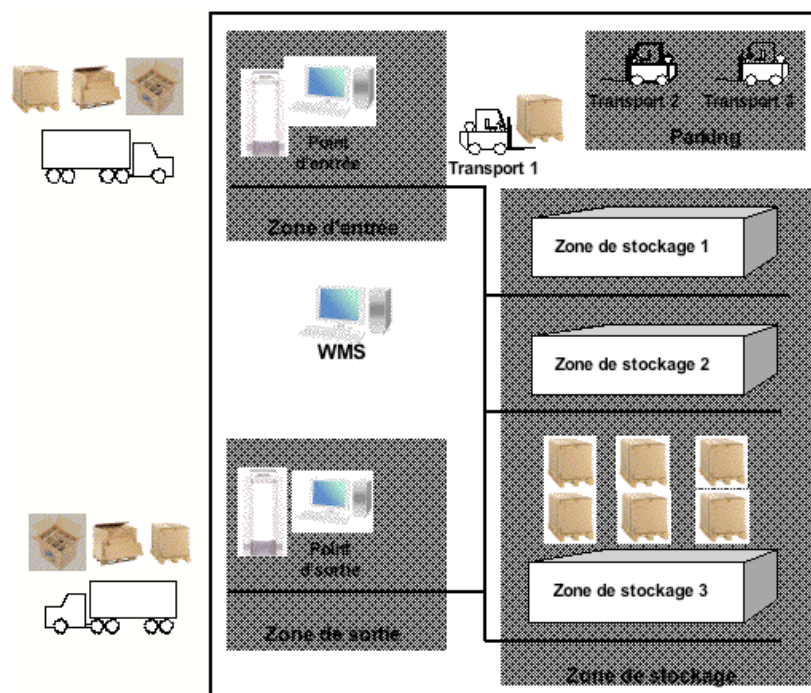


Figure 87 : Exemple de disposition des ressources dans un entrepôt.

La Figure 87 présente un exemple de disposition des ressources dans un entrepôt. On observe une zone d'entrée, une zone de stockage, une zone de sortie, un WMS (*Warehouse Management System*) [Mason et al., 2003] et un parking de chariots de transport.

Nous rappelons que nous proposons un **objet communicant passif** constitué d'une partie physique (l'objet physique), une étiquette électronique, un dispositif (dv) UPnP et un lecteur RFID intégré à ce dernier. Dans ce cas, l'objet a un rôle de **fournisseur de services**. Ce cas représente le premier cas proposé dans notre méthodologie (Chapitre 4). Ainsi, les acteurs de l'entrepôt peuvent interagir avec l'objet communicant pour lui demander des services à distance.

Dans le cas d'un **objet communicant actif**, l'objet est constitué d'une partie physique, d'une étiquette électronique, d'un dispositif (cp) UPnP (agissant comme dispositif et point de contrôle) et d'un lecteur RFID intégré dans le dispositif. De cette façon on obtient un objet qui peut agir comme **demandeur ou fournisseur de services**. Cet objet est représenté dans le deuxième cas de la proposition méthodologique (Chapitre 4).

Dans le point suivant, nous allons expliquer les acteurs génériques d'un entrepôt, dont les objets communicants passifs et actifs.

5.2 Architecture de Services proposée pour un entrepôt

Dans la Figure 88 nous illustrons l'architecture proposée pour gérer les interactions entre les acteurs de l'entrepôt, dont les produits communicants. La partie physique des produits communicants est représentée par trois familles d'objets physiques : les palettes, les boîtes de produits, et les produits individuels. Les palettes sont composées par des boîtes de produits physiques. Les boîtes sont composées par des produits physiques. Les objets physiques sont équipés d'étiquettes électroniques RFID.

Il faut noter que la « granularité » de l'interaction avec les produits peut se limiter aux boîtes de produits, voire aux palettes, pour correspondre aux usages industriels actuels d'étiquetage électronique ou « taggage » des produits tels qu'on les rencontre dans les applications industrielles de grande distribution. Mais notre approche ne distingue pas de limitation en ce domaine.

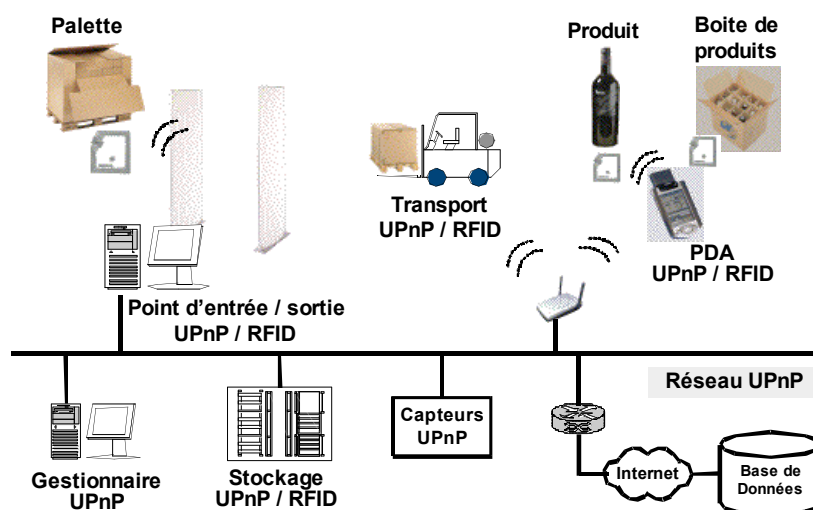


Figure 88 : Architecture de Services proposée pour un entrepôt.

Les dispositifs qui peuvent interagir avec les objets physiques sont les suivants :

- **Point d'entrée / sortie UPnP / RFID** : Des dispositifs UPnP équipés d'un lecteur RFID situés à l'entrée et à la sortie de l'entrepôt permettant l'identification automatique des objets

physiques arrivant ou sortant de l'entrepôt. Au moment de l'identification d'un objet physique, l'objet communicant et ses services sont créés dans le réseau de services ambiants UPnP.

- **Transport UPnP / RFID** : Des moyens de transports capables d'identifier automatiquement les objets physiques (palettes, boîtes et produits physiques), et de cette façon, permettre la création d'un objet communicant dans le réseau UPnP. Les moyens de transport effectuent les déplacements des objets physiques à l'intérieur de l'entrepôt.
- **PDA UPnP / RFID** : Des dispositifs mobiles UPnP équipés d'un lecteur RFID permettant l'identification des objets physiques stockés dans l'entrepôt, et en plus, la création d'un objet communicant capable de s'exprimer virtuellement sur un PDA (*Personal Data Assistant*).
- **Gestionnaire UPnP** : Entité informatique développant des fonctionnalités UPnP qui a une vision globale du flux physique et informationnel échangé entre les acteurs dans l'entrepôt, afin d'optimiser le mouvement d'objets physiques. En anglais, un système doté de telles caractéristiques est nommé *Warehouse Management System (WMS)*.
- **Stockage UPnP / RFID** : Des dispositifs UPnP situés dans les zones de stockage de l'entrepôt permettant l'identification automatique des objets physiques qui seront stockés. Lorsqu'un objet physique est identifié automatiquement par un dispositif, l'objet communicant est créé.
- **Capteurs UPnP** : Des dispositifs UPnP capables de capter le niveau de température, d'humidité... dans l'entrepôt, et spécifiquement, dans les différentes zones de stockage des objets physiques afin de surveiller l'environnement des objets physiques.
- **Bases de données** : Des bases de données accessibles à distance, capables d'être reconnues dans le réseau ambiant UPnP de l'entrepôt, dans le but d'obtenir de l'information complémentaire associée aux objets physiques. C'est-à-dire, chaque fournisseur d'objets physiques aura sa propre base de données afin de fournir l'information augmentée sur les objets physiques si nécessaire.

5.2.1 Facteurs déterminant les services d'un produit ou objet communicant

5.2.1.1 Services dépendant de la famille et type d'objet physique

Les services seront dépendants de la famille des objets physiques. Les familles d'objets que nous avons considérées sont : palettes, boîtes de produits ou produits individuels. Pour chaque famille d'objets, nous pouvons considérer différents types d'objets. Ainsi, pour la famille d'objets « palettes », il est possible de considérer différents types de palettes : palettes homogènes, palettes hétérogènes, ... Pour la famille d'objets « boîtes de produits » il est possible de classer les boîtes par leur contenu, leur taille, etc. La famille d'objets « produits » peut être décomposée par types de produits : produits de bureau, produits électroniques, produits laitiers, produits sensibles, ...

5.2.1.2 Services dépendants du domaine d'interaction

Nous avons établi, dans le chapitre 4, que les domaines d'interaction pour les acteurs d'un système sont déterminés par la phase du cycle de vie d'un objet communicant. Donc, pour chaque phase du cycle de vie, il faut étudier et analyser soigneusement les domaines d'interactions dans lesquelles un objet communicant va interagir avec d'autres acteurs communicants. Dans le cas d'un entrepôt, nous avons identifié quatre espaces physiques génériques dans lesquels les objets physiques peuvent interagir :

- zone d'entrée des objets physiques à l'entrepôt ;
- zone transport d'objets physiques ;
- zone stockage physique des objets ;
- zone sortie des objets physiques de l'entrepôt.

Chacun de ces espaces d'interactions est associé à l'état de l'objet.

Dans un entrepôt nous avons identifié trois types d'états des objets physiques :

- l'état physique de l'objet (normal, endommagé, détruit) ;
- l'état opérationnel (opération à développer avec l'objet : transporter, stocker, replacer, ...) ;
- l'état de satisfaction des exigences de stockage de l'objet (accomplissement des variables critiques associés à l'objet : date d'expiration, température, humidité, ...).

La notion spatiale (espace géographique), la notion d'état de l'objet et la notion temporelle définent le domaine d'interaction des objets physiques dans un entrepôt. En conséquence, les interactions que l'objet physique pourra réaliser dans l'entrepôt seront déterminées par ses domaines d'interaction. Par exemple, un objet physique en état normal, avec une date d'expiration inférieure à la date actuelle, étant entré à l'entrepôt, et attendant un moyen de transport, devra forcément être dirigé vers un certain endroit de stockage dans l'entrepôt.

5.2.1.3 Services dépendant du profil de l'acteur de l'entrepôt

Comme acteurs dans un entrepôt qui peuvent avoir la nécessité réelle d'interagir avec l'objet physique nous distinguons : le gestionnaire de l'entrepôt, les moyens de transport et les utilisateurs humains équipés de dispositifs PDA. Chacun des acteurs indiqués, pourra accéder aux services du produit communicant en fonction de son profil. Les produits communicants pourront offrir des services spécifiques aux acteurs si ces derniers ont le droit de le faire. Par exemple, des informations détaillées sur la composition physique, chimique, ... d'un objet pourront être présentées à un utilisateur humain, au travers d'un dispositif PDA UPnP, si ses droits d'accès le permettent. En particulier, l'information augmentée de l'objet physique, disponible sur une base de données distante et accessible au travers d'une interface utilisateur (PDA, PC,...) en fonction des droits des acteurs de l'entrepôt.

5.3 Classification des services offerts par les acteurs de l'entrepôt

Nous avons distingué cinq **classes de services** offertes par les acteurs de l'entrepôt : identification d'un acteur, description d'un acteur, état d'un acteur, traçabilité d'un acteur et information augmentée d'un acteur. L'application de chaque classe de services à chacun des acteurs define les **types de services** offerts par un acteur dans l'entrepôt.

5.3.1 Classe de service « Identification d'un acteur »

Cette classe de service permet d'identifier un acteur qui est connecté au réseau UPnP de l'entrepôt. L'identification d'un acteur permettra à un demandeur de services de rechercher un acteur spécifique dans l'entrepôt à un moment donné afin d'interagir avec lui. Dans le cas d'un objet physique, nous proposons d'utiliser le code EPC (*Electronic Product Code*) permettant d'identifier l'entreprise productrice de l'objet, le type d'objet (EPC) et l'objet physique avec un numéro de série. L'identification des autres acteurs de l'entrepôt nécessite l'affectation de codes uniques d'identification.

5.3.2 Classe de service « Description d'un acteur »

Cette classe de service permet aux demandeurs de service d'obtenir une description détaillée de chaque acteur connecté au réseau ambiant UPnP de l'entrepôt afin de connaître ses caractéristiques de base. En particulier, la description d'un objet physique permet d'informer les demandeurs de services sur ses conditions de stockage dans l'entrepôt. Les conditions de stockage sont représentées pour un ensemble de données mémorisées dans l'étiquette électronique attachée à l'objet physique. Ces données peuvent être : la température et l'humidité requise pour le stockage de l'objet, le poids de l'objet, les dimensions de l'objet, ... Dans le cas d'un objet physique composé par d'autres objets (palette et boîte), les conditions de stockage de l'objet doivent représenter convenablement l'ensemble d'objets qu'il contient.

La description du gestionnaire de l'entrepôt incorpore les conditions de stockages supportées par l'entrepôt. La description d'un moyen de transport doit contenir ses caractéristiques de déplacement telles que : la vitesse, l'accélération, la capacité de transport, le temps pour prendre et déposer un objet et la consommation d'énergie. La description d'une zone de stockage particulière doit contenir des informations telles que : la capacité de stockage, les conditions d'opération de la zone de stockage (par exemple le poids et les dimensions permises (intervalle) pour stocker les objets). La description d'un capteur UPnP permet d'informer les intéressés du détail des capacités de perception développées par le capteur telles que le type de variables supportées (température, bruit, lumière, ...), la précision des mesures et le temps d'actualisation des mesures). La description d'un utilisateur PDA doit contenir le type d'utilisateur et ses droits d'accès aux services des acteurs de l'entrepôt.

En bref, la description des acteurs permet aux demandeurs de services de connaître les valeurs des variables qui les décrivent en détail afin de choisir correctement l'acteur le mieux adapté à ses besoins.

5.3.3 Classe de service « Etat d'un acteur »

Cette classe de service permet d'informer les demandeurs de service de l'état d'un acteur UPnP à un moment donné dans l'entrepôt. Dans notre cas, un état représente un attribut particulier d'un acteur à un moment donné (comme par analogie à une photographie) afin de discrétiser (réduire / simplifier) l'univers de possibilités comportementaux qui décrivent un acteur dans l'entrepôt. Dans le cas d'un objet physique, nous distinguons l'état physique de l'objet (normal, endommagé, ...), l'état opérationnel de l'objet dans l'entrepôt (stocké, en mouvement, ...) et l'état de satisfaction des exigences de l'objet (température satisfaite, humidité satisfaite, ...) dans l'endroit où il se trouve.

Dans le cas d'un moyen de transport, les états peuvent représenter la disponibilité de la ressource, l'utilisation spécifique de la ressource à un moment donné (déplacement des objets physiques, emplacement des objets physiques, maintenance, ...). Pour un acteur PDA, l'état permet de refléter, indirectement, l'activité qui est réalisée par l'utilisateur à un moment donné. Dans le cas d'un capteur, l'état indique si le capteur opère correctement dans l'entrepôt. Les états associés aux zones de stockage permettent d'informer la disponibilité de la zone de stockage, le niveau d'utilisation de la zone, la variété et la composition des objets présents dans la zone de stockage. Pour le gestionnaire de l'entrepôt, l'état reflète la disponibilité de stockage de l'entrepôt, le niveau d'utilisation de l'entrepôt, le niveau de qualité service offert aux produits communicants dans l'entrepôt.

5.3.4 Classe de service « Traçabilité d'un acteur »

Cette classe de service permet aux demandeurs de service d'accéder aux données de traçabilité associées à un acteur dans l'entrepôt. Afin de gérer la traçabilité des objets physiques, il faut connaître le domaine fonctionnel précédent (Où était l'objet ?), actuel (Où est l'objet ?), et suivant (Où va l'objet ?) de chaque produit communicant dans l'entrepôt. De plus, il est nécessaire de connaître la localisation géographique, représentée par les coordonnées x , y et z , de tous les acteurs de l'entrepôt, dont les produits communicants, afin de les suivre à tout moment. Ceci permet d'identifier le domaine géographique de l'acteur de l'entrepôt.

Le gestionnaire de l'entrepôt est le responsable de l'enregistrement des événements, des changements d'états, et des opérations réalisées sur chaque acteur de l'entrepôt.

5.3.5 Classe de service « Information augmentée de l'acteur »

Cette classe de service regroupe les fonctionnalités des acteurs de l'entrepôt qui offrent une information augmentée ou enrichie aux demandeurs de services dans le réseau UPnP. L'information augmentée sera obtenue, par un demandeur de service, en fonction de son profil.

- A. **Types de service « Information augmentée » d'un produit communicant :** L'information augmentée des produits communicants sera obtenue grâce à une base de données accessible à distance (par exemple chez le producteur). Comme exemple d'information augmentée, nous citons la description technique du produit communicant, le manuel d'utilisation du produit communicant, la procédure de recyclage du produit communicant.
- B. **Types de service « Information augmentée » des moyens de transport :** Les services d'information augmentée offerts par les moyens de transport sont associés à l'évaluation des demandes de transports effectués par les objets communicants actifs (produits, boîtes, palettes). Cette évaluation consiste à étudier les exigences réalisées par un objet physique (représentées par les conditions de stockage et son état actuel) et la charge de travail qu'a le moyen de transport afin d'émettre une réponse en indiquant, dans un premier temps, la disponibilité pour réaliser la tâche (oui / non), et dans un deuxième temps, une proposition pour réaliser la tâche demandée (transport).
- C. **Types de service « Information augmentée » des capteurs de l'entrepôt :** L'information augmentée des capteurs UPnP permet aux demandeurs de services dans l'entrepôt d'obtenir les valeurs des variables mesurées par les capteurs (surveiller l'environnement), par exemple la température et l'humidité de l'environnement de stockage.

D. **Types de service « Information augmentée » du gestionnaire de l'entrepôt** : Les services d'informations augmentées offerts par le gestionnaire de l'entrepôt sont :

- D.1 **Type de service « Information augmentée : Disponibilité de stockage »** : Dans le cas d'un objet physique, le gestionnaire de l'entrepôt lui permet de s'informer sur la disponibilité de stockage de l'entrepôt, la qualité de service offerte par l'entrepôt (représentée par un ensemble d'indicateurs tels que le niveau d'accomplissement des conditions de stockage demandées par l'objet physique, le temps d'attente prévu avant d'être correctement stocké, la sécurité offerte, ...).
- D.2 **Type de service « Information augmentée : Traçabilité des acteurs »** : Le gestionnaire de l'entrepôt offre des services pour gérer la traçabilité des acteurs suivants : objets physiques, transports UPnP / RFID, zones de stockage UPnP / RFID et utilisateurs (PDA) à l'intérieur de l'entrepôt. Ceci dans le but de connaître les actions réalisées et en cours de réalisation par chacun d'eux. Il est souhaitable d'enregistrer à tout moment la relation entre l'objet physique, une ressource interagissant avec l'objet physique (transport, utilisateur PDA), la localisation spatiale, le temps et les valeurs réels des variables de stockage associées aux objets physiques.
- D.3 **Type de service « Information augmentée : Gestion des moyens de transport »** : Le gestionnaire de l'entrepôt permet aux objets communicants actifs d'obtenir une priorité pour utiliser un moyen de transport à l'intérieur de l'entrepôt. Cette priorité est calculée en fonction des variables décrivant l'objet physique et les conditions réelles de stockage associées à un objet à un moment donné. Par exemple, si un objet a besoin d'une température de X degrés Celsius, et si l'endroit où l'objet est présent, a une température en dehors de l'intervalle d'acceptation, l'objet aura une haute priorité afin d'être replacé dans un endroit adapté à ses besoins. Il est important d'indiquer que la priorité attribuée à un objet est dynamique et devra être calculée chaque fois que l'état de l'objet est modifié. Egalement, le gestionnaire offre des services associés au routage des transports (optimisation des déplacements et minimisation des conflits entre ressources).
- D.4 **Type de service « Information augmentée : Maintenance des acteurs de l'entrepôt »** : Le gestionnaire de l'entrepôt permet aux acteurs de l'entrepôt de demander un service de maintenance si nécessaire.
- D.5 **Type de service « Information augmentée : Communication vers l'extérieur »** : Le gestionnaire de l'entrepôt permet aux objets physiques de communiquer ses changements d'état à des entités externes. Par exemple, l'objet peut envoyer des messages en indiquant son état et ses conditions de stockage réelles aux clients externes en utilisant des services adaptés offerts par le gestionnaire de l'entrepôt grâce à sa vision globale du système et des entités participant à la chaîne logistique.

5.4 Relations entre les acteurs de l'entrepôt par les services

Dans la Figure 89 nous illustrons les relations entre les acteurs de l'entrepôt au travers des services que chacun met à disposition des autres dans le réseau ambiant de services. Chaque flèche représente une demande de service réalisée par un demandeur de service vers un fournisseur de service. Ainsi, la pointe de la flèche indique le réceptionnaire de la demande de service (fournisseur de service). On peut distinguer deux attracteurs d'offres et de demandes de services dans la figure : l'objet communicant actif / passif et le gestionnaire de l'entrepôt. D'une part, l'objet communicant actif réalise des demandes de services vers le gestionnaire de l'entrepôt, les moyens de transport et les capteurs. En plus, l'objet communicant passif reçoit des demandes de services depuis les moyens de transport, le gestionnaire de l'entrepôt et l'utilisateur PDA. D'autre part, le gestionnaire de l'entrepôt réalise des demandes de services vers les moyens de transport, les capteurs et les utilisateurs PDA. En plus, le gestionnaire de l'entrepôt reçoit des demandes de services depuis les moyens de transport, les capteurs, les zones de stockage et les utilisateurs PDA.

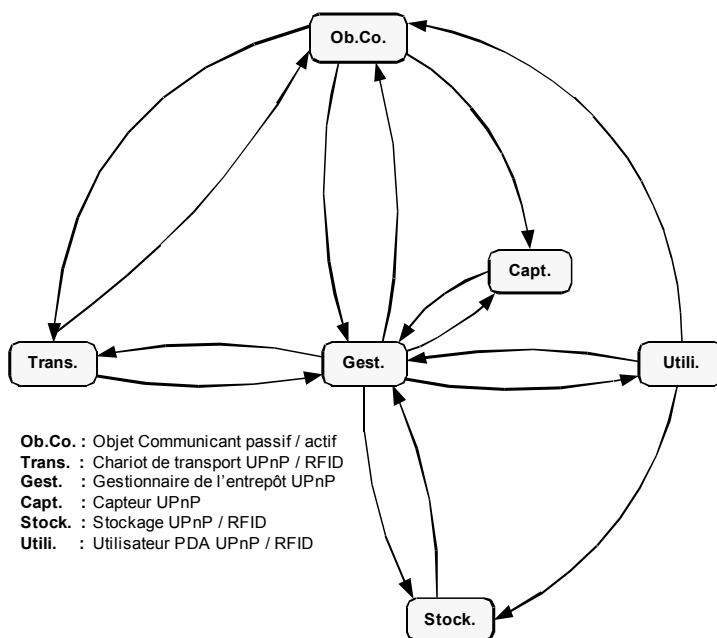


Figure 89 : Diagramme de relations entre les acteurs de l'entrepôt.

Dans le Tableau 10 nous représentons les relations par les services entre les acteurs de l'entrepôt. Ce tableau exprime les relations graphiques montrées dans la Figure 89. Les cellules grises indiquent qu'il n'y a pas une relation significative entre les acteurs de l'entrepôt.

	Objet Communicant	Transport	Gestionnaire	Capteurs	Stockage	Utilisateur
Objet Communicant		Tous les types de services de l'objet communicant.	Tous les types de services de l'objet communicant.			Tous les types de services de l'objet communicant.
Transport	Tous les types de services d'un moyen transport.		Tous les types de services d'un moyen de transport (sauf l'évaluation de transport)			
Gestionnaire	Disponibilité de stockage, qualité de service de l'entrepôt, priorité de transport, accepter stockage, demander un autre endroit de stockage, communiquer avec des entités externes.	Routage du transport dans l'entrepôt, maintenance pour un transport.		Maintenance pour un capteur.	Maintenance pour une zone de stockage.	Traçabilité des acteurs de l'entrepôt.
Capteurs	Tous les types de services d'un capteur.		Tous les types de services d'un capteur.			
Stockage			Tous les types de services d'une zone de stockage.			Tous les types de services d'une zone de stockage.
Utilisateur			Tous les types de services d'un utilisateur.			

Tableau 10 : Relations entre acteurs par les services dans l'entrepôt.

Dans la section suivante nous allons approfondir les services demandés par l'objet communicant actif aux acteurs de l'entrepôt (colonne 1 du Tableau 10).

5.4.1 Relation entre un objet communicant actif et le gestionnaire de l'entrepôt UPnP

Notre intérêt est de montrer le rôle de demandeur de service du produit communicant dans l'entrepôt au moyen des relations entre l'objet communicant actif et le gestionnaire de l'entrepôt. Le diagramme d'utilisation UML représenté dans la Figure 90 illustre l'utilisation du gestionnaire de l'entrepôt par un objet physique communicant. Ces cas d'utilisation permettent de représenter les situations dans lesquelles un objet physique communicant agit comme demandeur de services vers le gestionnaire de l'entrepôt. L'objet communicant actif utilise le gestionnaire de l'entrepôt pour obtenir la faisabilité de stockage, connaître la qualité de service offerte par l'entrepôt afin d'être stocké, informer l'entrepôt de la décision prise par rapport au stockage proposé, demander un autre endroit de stockage au moyen de la communication entre gestionnaires de stockage, communiquer ses variables d'état à des clients externes (messagerie électronique, ...) grâce à la vue globale qu'a le gestionnaire (données de clients, données de traçabilité, ...).

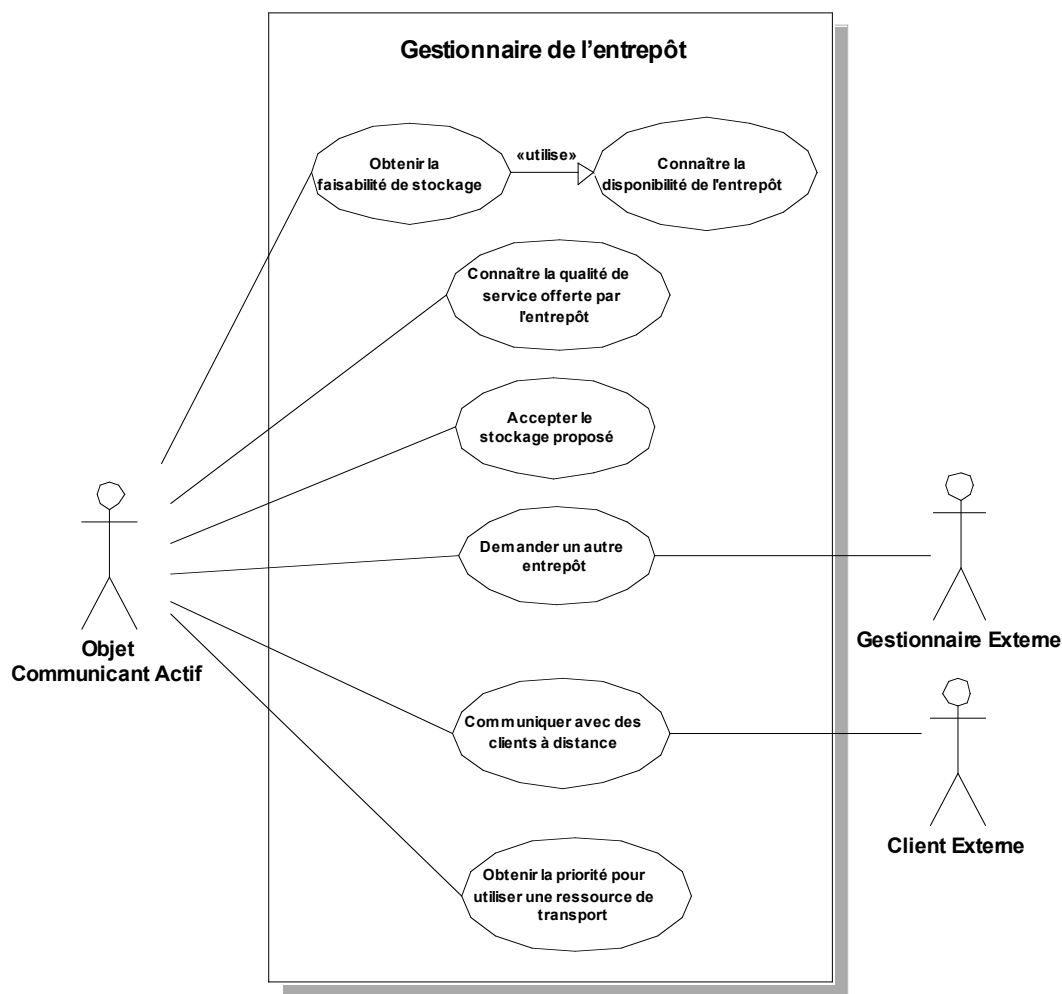


Figure 90 : Diagramme de cas d'utilisation UML décrivant l'usage du gestionnaire de l'entrepôt par l'objet communicant actif.

En considérant les caractéristiques dynamiques des objets physiques, il est possible d'associer une priorité de traitement ou de travail à chaque objet physique afin de rationaliser l'utilisation d'une ressource particulière dans un environnement donné. Dans notre cas, l'intérêt est d'associer à chaque objet une priorité de transport dans l'entrepôt afin de transporter d'abord les objets prioritaires (urgences, dates critiques, ...) et ensuite les objets avec une priorité normale voire plus basse. Dans ce sens, le diagramme de la Figure 90 illustre aussi un objet communicant actif en relation avec le gestionnaire de l'entrepôt afin d'obtenir la priorité pour utiliser une ressource de transport dans l'entrepôt.

Le diagramme de séquence UML représenté dans la Figure 91 illustre le processus dynamique d'affectation d'une priorité pour un demandeur de services. Une fois que le demandeur de service a besoin d'utiliser une ressource critique (très demandée), il demande une priorité d'usage au gestionnaire de l'entrepôt. Afin de calculer une priorité initiale pour l'objet physique le gestionnaire a besoin de connaître les variables décrivant l'état physique de l'objet et l'état de satisfaction actuel des conditions de stockage demandées. Pour cela, il demande ces variables à l'objet communicant actif. Avec la réponse de l'objet le gestionnaire calcule la priorité demandée. Comme l'état physique de l'objet et l'état de satisfaction de stockage peuvent changer dans le temps, le gestionnaire, souscrit à ces variables, reçoit les mises à jour et recalcule la priorité. Si la nouvelle priorité est différente de la précédente il envoie la nouvelle priorité au demandeur de services. Celui-ci pourra annuler le contrat

établi précédemment avec le fournisseur de service et pourra également générer un nouveau compromis avec le fournisseur de service pour exécuter une tâche donnée.

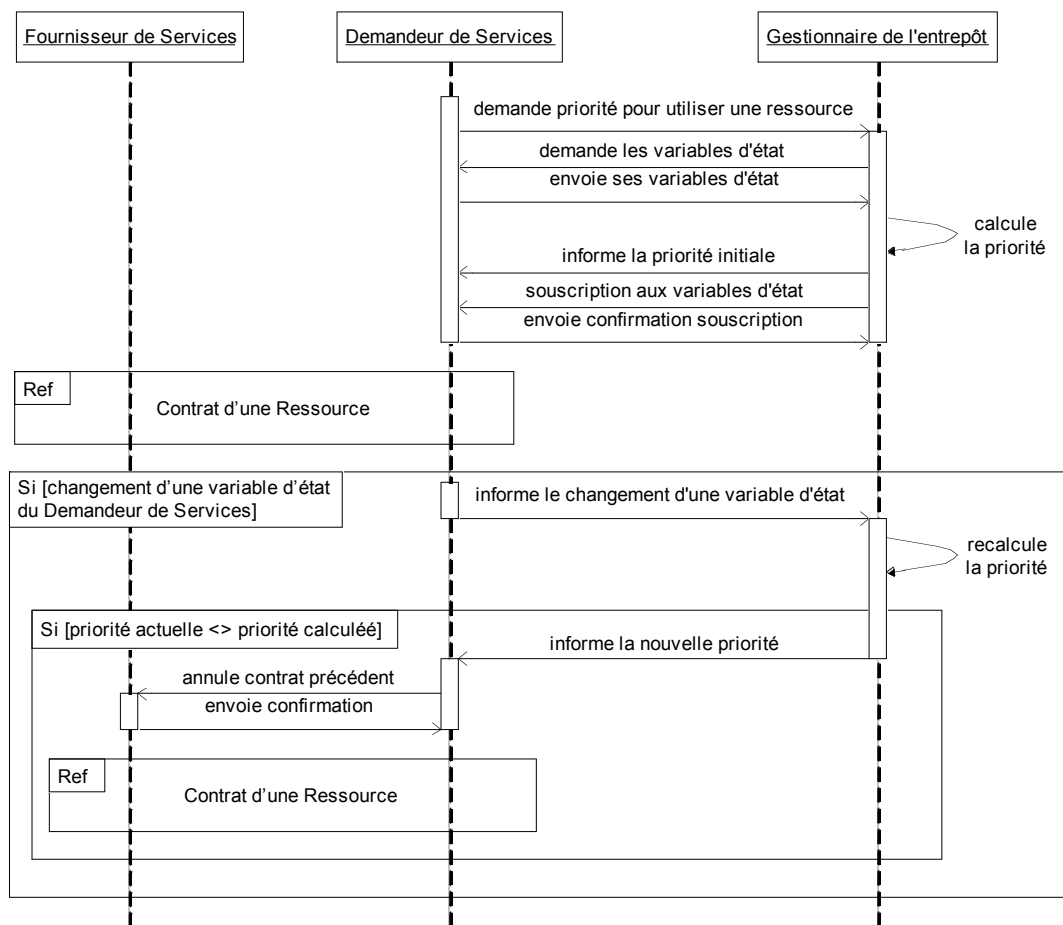


Figure 91 : Diagramme de séquence UML décrivant le mécanisme de gestion de priorités

5.4.2 Relation entre un objet communicant actif et les moyens de transports UPnP

L'objet communicant actif, étant un demandeur de services, interagit avec les moyens de transport UPnP afin de gérer ses déplacements dans l'entrepôt. Dans le diagramme de séquence UML représenté dans la Figure 92 nous illustrons le mécanisme général employé par un objet communicant actif afin de choisir une ressource parmi plusieurs candidats. La figure montre que le processus commence avec la demande d'un service (utilisation d'une ressource, transport, ...) envoyée par le demandeur de service (objet communicant actif) vers les fournisseurs de services disponibles (**m**) dans le réseau ambiant IP UPnP. Le fournisseur de service répond en demandant les caractéristiques souhaitées (conditions de stockage, conditions de transport, ...) au demandeur de service. Les fournisseurs qui offrent le service demandé et qui peuvent répondre à la demande (**p**) informeront le demandeur de service de leur disponibilité. En plus, les fournisseurs de services indiqueront une proposition d'exécution du service (rendez-vous, temps d'exécution de la tâche, ...). Ainsi, le demandeur de services pourra choisir le fournisseur de service le mieux adapté à ses besoins. Ensuite, il enverra l'acceptation de la proposition au fournisseur de service choisi. Ceci représente un contrat entre le demandeur et le fournisseur de service pour exécuter le service à un moment et un endroit donné. Après l'exécution d'un service, le fournisseur de services pourra envoyer un message au demandeur de service en indiquant le résultat de l'action demandée. Ce mécanisme est inspiré par le protocole d'interaction *Contract Net* standardisé par la FIPA [FIPA, 2002].

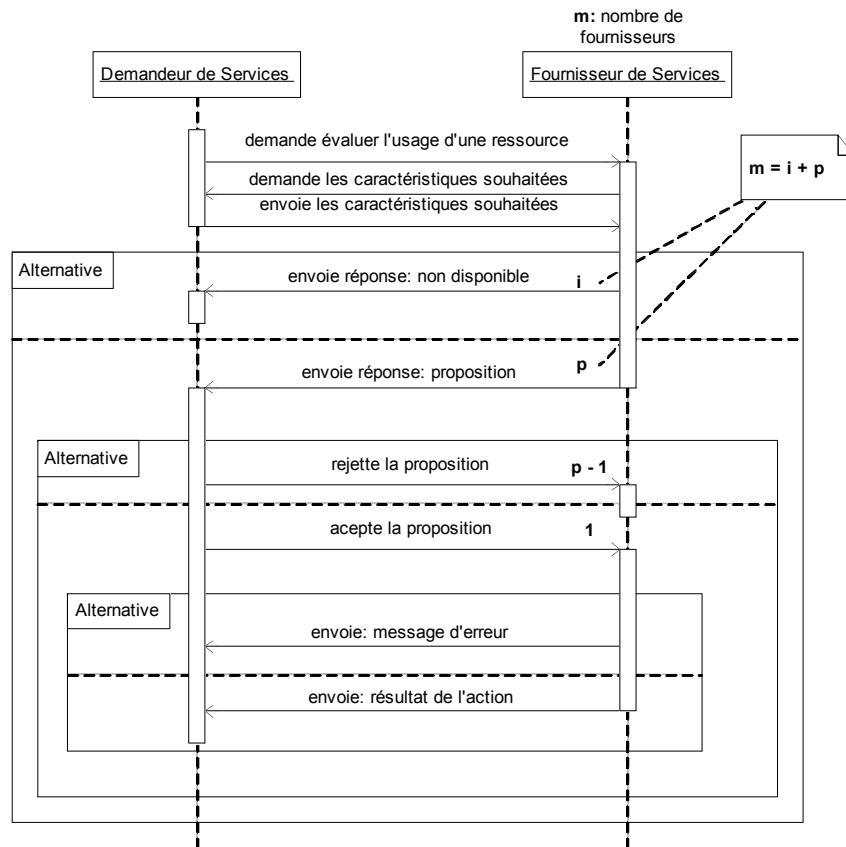


Figure 92 : Diagramme de séquence UML décrivant le mécanisme pour établir un contrat de travail entre un demandeur et un fournisseur de services.

La Figure 93 illustre un diagramme de cas d'utilisation UML représentant un objet communicant actif qui peut utiliser une ressource de transport (chariot) afin de demander son utilisation, confirmer son élection et utilisation, annuler un contrat d'utilisation créé auparavant.

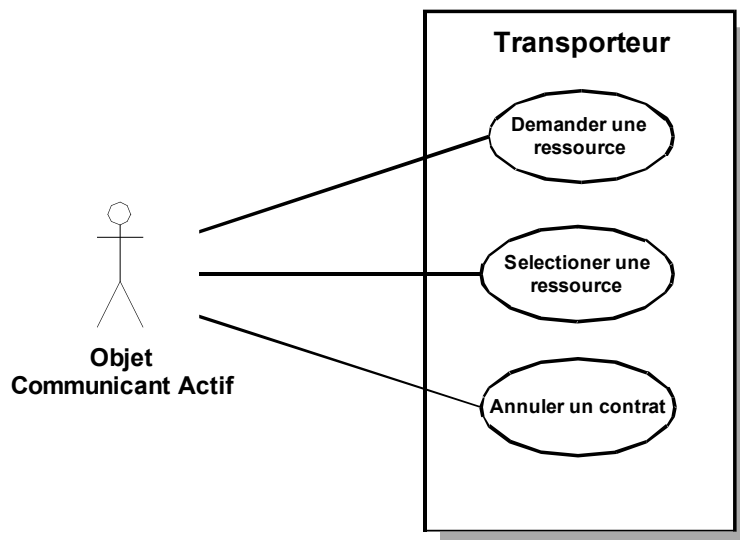


Figure 93 : Diagramme de cas d'utilisation UML

5.4.3 Relation entre un objet communicant actif et capteur UPnP

Dans le cas où les objets communicants ne sont pas équipés de capteurs sensitifs, l'utilisation de capteurs indépendants dans l'entrepôt permet à l'objet de mieux connaître son contexte de travail. En effet, la relation entre un objet communicant actif et un ensemble de capteurs permet d'apporter à l'objet communicant des capacités de perception étendues sur son environnement. Ainsi, les capteurs répartis dans l'entrepôt sont employés comme une extension sensorielle des objets physiques.

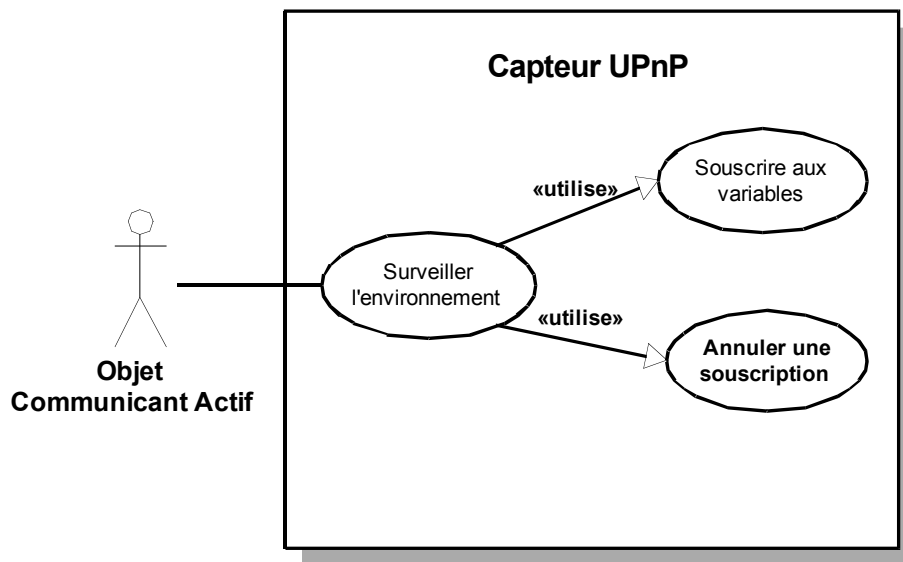


Figure 94 : Diagramme de cas d'utilisation UML

Dans le diagramme de cas d'utilisation de la Figure 94, nous illustrons de quelle façon l'objet communicant actif utilise un capteur UPnP dans un entrepôt. On observe que l'objet communicant utilise un capteur pour surveiller son environnement afin de détecter des situations préjudiciables pour son intégrité physique et lui permettre de réagir à temps. Pour cela, l'objet communicant actif peut souscrire aux changements de ces variables. Une souscription sera annulée quand l'objet n'aura pas besoin de l'information apportée par un capteur donné.

5.5 Définition des services offerts par les produits communicants

Dans cette partie nous allons détailler les types de services qu'un objet physique peut offrir aux demandeurs de services dans un entrepôt. Les types de services contiennent un ensemble de services qui sont créés à partir des données stockées sur l'étiquette électronique attachée à l'objet physique.

5.5.1 Type de service « Identification d'un produit communicant »

Dans le Tableau 11, nous signalons la liste de services permettant l'identification d'un objet communicant dans l'entrepôt. Ici, le service clé est le service d'identification EPC permettant l'identification d'un objet physique d'une façon unique. De plus, le numéro de série de l'étiquette sera utilisé pour identifier chaque étiquette, associée à un objet communicant, de façon individuelle. Ainsi, nous pouvons gérer le cycle de vie de l'étiquette en la réutilisant plusieurs fois en fonction du cycle de vie des objets physiques. Afin de détailler la composition d'un objet physique, nous avons inclus le nombre d'objets à l'intérieur d'un objet (nombre d'objets dans une boîte et sur une palette).

Service	Objectif du Service
Service SNR	Obtenir le numéro de série (<i>Serial Number</i>) de l'étiquette attachée à l'objet physique.
Service EPC	Obtenir le EPC (<i>Electronic Product Code</i>) de l'objet physique.
Service Composition d'une Boite	Obtenir et modifier le nombre de produits contenus dans une boîte de produits.
Service Composition d'une Palette	Obtenir et modifier le nombre de produits et boîtes contenus dans une palette.

Tableau 11 : Liste des services pour identifier un produit communicant

Le Tableau 12 montre la liste de variables que nous devons considérer afin d'identifier un objet communicant dans l'entrepôt. Les variables représentant le nombre d'objets contenus à l'intérieur d'un objet physique doivent être de nature événementielle afin d'informer des changements d'état au long de son transport ou stockage. Le SNR est incorporé par le fabricant de l'étiquette. Sans considérer le SNR, nous devons réserver 128 bits de mémoire (soit 17 octets) sur l'étiquette électronique pour implémenter ce service.

Service	Variable	Description	Type	Bits sur le tag	Événementiel
SNR	SNR	Numéro de série de l'étiquette	String	64	Non
EPC	EPC_version	Version du code EPC	String	8	Non
EPC	EPC_Manager	Entreprise qui génère l'objet physique	String	28	Non
EPC	EPC_Class	Type d'objet physique	String	24	Non
EPC	EPC_Serial	Numéro sérialisé de l'objet physique	String	36	Non
Composition d'une Boite	Nr_products	Nombre de produits dans une boîte de produits	Entier	16	Oui
Composition d'une Palette	Nr_boites	Nombre de boîtes sur une palette	Entier	8	Oui
Composition d'une Palette	Nr_products	Nombre de produits dans une boîte de produits	Entier	16	Oui

Tableau 12 : Variables décrivant les services pour identifier un produit communicant.

Le Tableau 13 indique les actions permettant d'exécuter les services d'identification d'un objet physique. Une action de type « *Get* » implique une action de lecture (*output*) sur l'étiquette et une action de type « *set* » implique une action d'écriture sur l'étiquette (*input*).

Service	Action	Description de l'action	Variables	Orientation
SNR	getSNR	Obtenir le SNR de l'étiquette électronique.	SNR	<i>Output</i>
EPC	getEPC_Manager	Obtenir le code de l'entreprise qui a créé l'objet physique.	EPC_Manager	<i>Output</i>
EPC	getEPC_Class	Obtenir le code du type objet physique.	EPC_Class	<i>Output</i>
EPC	getEPC_Serial	Obtenir le code sérialisé de l'objet physique.	EPC_Serial	<i>Output</i>
Composition d'une Boite	getNr_produits	Obtenir le nombre de produits contenus dans une boite de produits.	Nr_produits	<i>Output</i>
Composition d'une Boite	setNr_produits	Modifier le nombre de produits contenus dans une boite de produits.	Nr_produits	<i>Input</i>
Composition d'une Palette	getNr_boites	Obtenir le nombre de boites contenues sur une palette.	Nr_boites	<i>Output</i>
Composition d'une Palette	getNr_produits	Obtenir le nombre de produits contenus dans une boite de produits.	Nr_produits	<i>Output</i>
Composition d'une Palette	setNr_boites	Modifier le nombre de boites contenues sur une palette.	Nr_boites	<i>Input</i>
Composition d'une Palette	setNr_produits	Modifier le nombre de produits contenus dans une boite de produits.	Nr_produits	<i>Input</i>

Tableau 13 : Actions permettant l'identification d'un produit communicant.

5.5.2 Type de service « Conditions de stockage d'un produit communicant »

Le Tableau 14 signale la liste de services permettant d'informer les demandeurs de services sur les conditions de stockage requises pour stocker correctement un objet physique dans un entrepôt. Ces services doivent incorporer, au moins, l'information sur les dimensions d'un objet physique, sa date d'expiration, ses conditions critiques de stockage (intervalle de température et d'humidité), et son niveau de fragilité.

Service	Objectif du Service
Dimensions	Obtenir les dimensions d'un objet physique
Date d'expiration	Obtenir la date d'expiration d'un objet physique.
Température	Obtenir l'intervalle de température supporté par l'objet physique.
Humidité	Obtenir l'intervalle d'humidité supporté par l'objet physique.
Fragilité	Obtenir le niveau de fragilité de l'objet physique.

Tableau 14 : Liste de service décrivant les conditions de stockage d'un objet physique.

Le Tableau 15 identifie et décrit les variables associées à chacun des services signalés. On doit disposer de 73 bits, c'est-à-dire, nous devons réserver 10 octets sur l'étiquette électronique afin d'implémenter les services indiqués. Dans ce cas, toutes les variables identifiées sont non événementielles parce que les conditions de stockage définies pour un objet physique sont invariables.

Service	Variable	Description	Type	Valeurs permissibles	Bits sur le tag	U. mesure	Événementiel
Dimensions	DimX DimY DimZ	- Longueur de l'objet physique - Largeur de l'objet physique. - Hauteur de l'objet physique.	Entier Entier Entier	-	8 + 8 + 8	cm	Non
Date d'expiration	ED_Day ED_Month ED_Year	- Jour d'expiration de l'objet physique. - Mois d'expiration de l'objet physique. - Année d'expiration de l'objet physique.	Entier Entier Entier	-	5 + 4 + 7	-	Non
Température	Temp_min Temp_max	Température minimale et maximale supportée par l'objet physique.	Entier	-	8 + 8	°C	Non
Humidité	MinHum MaxHum	Humidité minimale et maximale supportée par l'objet physique.	Entier Entier	-	8 + 8	%	Non
Fragilité	Fragility	Fragilité d'un objet physique.	Entier	[Très fragile, fragile, normal]	1	-	Non

Tableau 15 : Variables décrivant les conditions de stockage d'un objet physique.

Le Tableau 16 montre les actions qui sont associés aux services décrivant les conditions de stockage des objets physiques. Toutes les actions sont de type « *Get* ».

Service	Action	Description de l'action	Variables	Orientation
Dimensions	getDimensionX getDimensionY getDimensionZ	- Obtenir la longueur de l'objet physique. - Obtenir la largeur de l'objet physique. - Obtenir la hauteur de l'objet physique.	DIM_X DIM_Y DIM_Z	<i>output</i>
Date d'expiration	getExpiryDay getExpiryMonth getExpiryYear	Obtenir le jour, la date et l'année de la date d'expiration de l'objet physique.	ED_Day ED_Month ED_Year	<i>output</i>
Température	getTemperatureMin getTemperatureMax	Obtenir la température minimale et maximale supportée par l'objet physique.	Temp_min Temp_max	<i>output</i>
Humidité	getMinHumidity getMaxHumidity	Obtenir humidité minimale et maximale supportée par l'objet physique.	MinHum MaxHum	<i>output</i>
Fragilité	getFragility	Obtenir le niveau de fragilité associé à l'objet physique.	Fragility	<i>output</i>

Tableau 16 : Actions permettant d'obtenir les conditions de stockage d'un objet physique.

5.5.3 Type de Service « Etats d'un produit communicant »

Dans ce type de service, nous incorporons trois services. Un service pour représenter l'état physique de l'objet. Un service pour représenter l'état opérationnel de l'objet, c'est-à-dire, l'opération qu'il faut développer sur l'objet (transport, ...). Un service pour représenter l'état des conditions demandées par l'objet, c'est-à-dire, la relation entre les conditions de stockage de l'objet et les conditions réelles de stockage de l'objet. Le Tableau 17 montre les services identifiés.

Service	Objectif du Service
Etat physique de l'objet	Obtenir et modifier l'état physique d'un objet physique.
Etat opérationnel de l'objet	Obtenir et modifier l'état opérationnel d'un objet physique.
Etat des conditions demandées par l'objet	Obtenir et modifier l'accomplissement des conditions de stockage demandées par l'objet physique.

Tableau 17 : Liste de services servant à informer et modifier l'état d'un produit communicant.

Le Tableau 18 montre les variables qui sont associées aux services signalés. Les variables sont de type événementiel. Ceci permet d'informer les demandeurs de services intéressés lorsqu'un changement dans une variable se produit. Il faut réserver 5 bits pour implémenter ces services.

Service	Variable	Description	Type	Valeurs permises	Bits sur le tag	Événementiel
Etat physique de l'objet	Pstate	Etat physique de l'objet à un instant donné.	Entier	[Normal, endommagé, détruit]	2	Oui
Etat opérationnel de l'objet	Ostate	Opération à développer sur l'objet physique.	Entier	[Transporter, Stocker, Remplacer, ..]	2	Oui
Etat des conditions de stockage	Cstate	Accomplissement des conditions de stockage demandées par l'objet.	Entier	[Oui, Non]	1	Oui

Tableau 18 : Variables décrivant les services pour identifier les états des produits communicants.

Le Tableau 19 indique les actions associées aux services gérant l'état d'un objet physique. Les actions permettent obtenir « *Get* » et changer « *Set* » les états définis des objets physiques.

Service	Action	Description de l'action	Variables	Orientation
Etat physique de l'objet	getPstate	Obtenir l'état physique de l'objet.	Pstate	<i>output</i>
Etat physique de l'objet	setPstate	Modifier l'état physique de l'objet.	Pstate	<i>input</i>
Etat opérationnel de l'objet	getOstate	Obtenir l'état opérationnel de l'objet.	Ostate	<i>output</i>
Etat opérationnel de l'objet	setOstate	Modifier l'état opérationnel de l'objet.	Ostate	<i>input</i>
Etat des conditions demandées de l'objet	getCstate	Obtenir l'accomplissement des conditions de stockage	Cstate	<i>output</i>

Tableau 19 : Actions sur les états des produits communicants.

5.5.4 Type de service « Traçabilité d'un produit communicant »

Le Tableau 20 montre les services que nous avons définis afin d'inclure les données de base pour tracer les objets physiques dans l'entrepôt. Pour cela, il faut identifier le domaine fonctionnel et géographique des objets physiques. Le domaine fonctionnel sera déterminé au moyen de la localisation fonctionnelle de l'objet. Dans ce cas, l'interaction objet – acteur de l'entrepôt permet d'inférer la fonctionnalité développée par / avec l'objet. Le domaine géographique sera déterminé par la localisation géographique de l'objet. Egalement, il faut connaître à tout moment la localisation fonctionnelle précédente et suivante de l'objet afin de garder le lien des opérations développées par / avec l'objet. Il est évident que cette information est insuffisante pour garder l'historique des transactions auxquelles l'objet participe. Pour cela, nous avons besoin du gestionnaire de l'entrepôt afin d'enregistrer les données critiques de l'objet pendant son stage dans l'entrepôt.

Service	Objectif du Service
Localisation fonctionnelle de l'objet physique	Obtenir et modifier la localisation fonctionnelle de l'objet physique
Localisation géographique de l'objet physique	Obtenir et modifier la localisation géographique de l'objet physique
Localisation fonctionnelle précédente de l'objet physique	Obtenir et modifier la localisation précédente de l'objet physique
Localisation fonctionnelle suivante de l'objet physique	Obtenir et modifier la localisation suivante de l'objet physique

Tableau 20 : Les services de domaine fonctionnel et géographique d'un produit communicant.

Dans le Tableau 21 nous listons les variables associées aux services signalés dans le tableau précédent. Il est évident que toutes les variables sont événementielles. Afin d'enregistrer les variables identifiées nous avons besoin de 50 bits de mémoire sur l'étiquette électronique de l'objet physique.

Service	Variable	Description	Type	Valeurs permises	Bits Sur le tag	U. mesure	Événementiel
Localisation fonctionnelle de l'objet physique	LocF	Localisation fonctionnelle de l'objet	Entier	[Entrée, Chariot, Etagère, Sortie]	2	-	Oui
Localisation géographique de l'objet physique	LocX LocY LocZ	Localisation géographique (X, Y, Z)	Entier Entier Entier	-	8 + 8 + 8	m.	Oui
Localisation fonctionnelle précédente de l'objet physique	LocP	Localisation précédente de l'objet physique.	Entier	-	16	-	Oui
Localisation fonctionnelle suivante de l'objet physique	LocS	Localisation suivante de l'objet physique.	Entier	-	16	-	Oui

Tableau 21 : Variables des domaines fonctionnel et géographique du produit communicant.

Dans le Tableau 22 nous montrons les actions que les demandeurs de service dans l'entrepôt peuvent invoquer sur l'objet. Ces actions sont de type « *Get* » et « *Set* ».

Service	Action	Description de l'action	Variables	Orientation
Localisation fonctionnelle de l'objet physique	getLocF	Obtenir la localisation fonctionnelle de l'objet.	LocF	Output
Localisation fonctionnelle de l'objet physique	setLocF	Modifier localisation fonctionnelle de l'objet.	LocF	Input
Localisation géographique de l'objet physique	getLocX getLocY getLocZ	Obtenir la localisation (X,Y,Z) de l'objet.	LocX LocY LocZ	Output
Localisation géographique de l'objet physique	setLocX setLocY setLocZ	Modifier la localisation (X,Y,Z) de l'objet.	LocX LocY LocZ	Input
Localisation fonctionnelle précédente de l'objet physique	getLocP	Obtenir la localisation précédente de l'objet.	LocP	Output
Localisation fonctionnelle précédente de l'objet physique	setLocP	Modifier la localisation précédente de l'objet.	LocP	Input
Localisation fonctionnelle suivante de l'objet physique	getLocS	Obtenir la localisation suivante de l'objet.	LocS	Output
Localisation fonctionnelle suivante de l'objet physique	setLocS	Modifier la localisation précédente de l'objet.	LocS	Input

Tableau 22 : Actions des services de traçabilité de l'objet communicant.

5.5.5 Type de service « Information augmentée d'un produit communicant »

Dans cette classe de service, nous représentons toute l'information technique et opérationnelle associée à l'objet qui est gérée par le fabricant ou le producteur de l'objet physique. Par exemple la date de fabrication de l'objet, la garantie de l'objet, etc. sont des informations qui peuvent être demandées par les acteurs de l'entrepôt à un moment donné dans un cas particulier. Pour cela, ce type d'information peut être stocké dans l'étiquette électronique de l'objet, en fonction de la capacité de mémoire de l'étiquette RFID, ou peut être demandé, par le demandeur de services, directement à une base de données chez le producteur de l'objet physique.

5.6 Interactions dans le système par les services

5.6.1 Entrée et stockage des produits communicants

La Figure 95 illustre les interactions entre acteurs réalisées lorsqu'un objet physique a besoin d'être stocké dans l'entrepôt. On observe qu'un camion avec de capacités de communication UPnP annonce sa présence à l'entrepôt à un moment donné afin d'indiquer son arrivée à l'entrepôt. Lorsqu'un objet physique a besoin de stockage, il contacte le gestionnaire de l'entrepôt afin d'évaluer la disponibilité de stockage de l'entrepôt. L'entrepôt demande à l'objet physique son identification (EPC) et ses variables décrivant ses conditions de stockage (dimensions, date d'expiration, température, humidité, fragilité, ...). L'objet communicant envoie les valeurs de ces variables au gestionnaire de l'entrepôt. Ensuite, le gestionnaire évalue la disponibilité de stockage et l'envoie à l'objet communicant. Si la réponse est positive, l'objet communicant peut demander plus d'informations sur son stockage, en demandant la qualité de service offerte par l'entrepôt en termes d'indicateurs tels que le temps d'attente, ... Avec cette information, l'objet peut décider de son stockage. Si l'objet accepte le stockage dans l'entrepôt, il demandera un moyen de transport dans l'entrepôt, en utilisant le mécanisme de gestion de priorités décrit dans la Figure 91. Le transport de l'objet s'effectuera selon le résultat obtenu. Le processus de transport sera expliqué dans le point suivant.

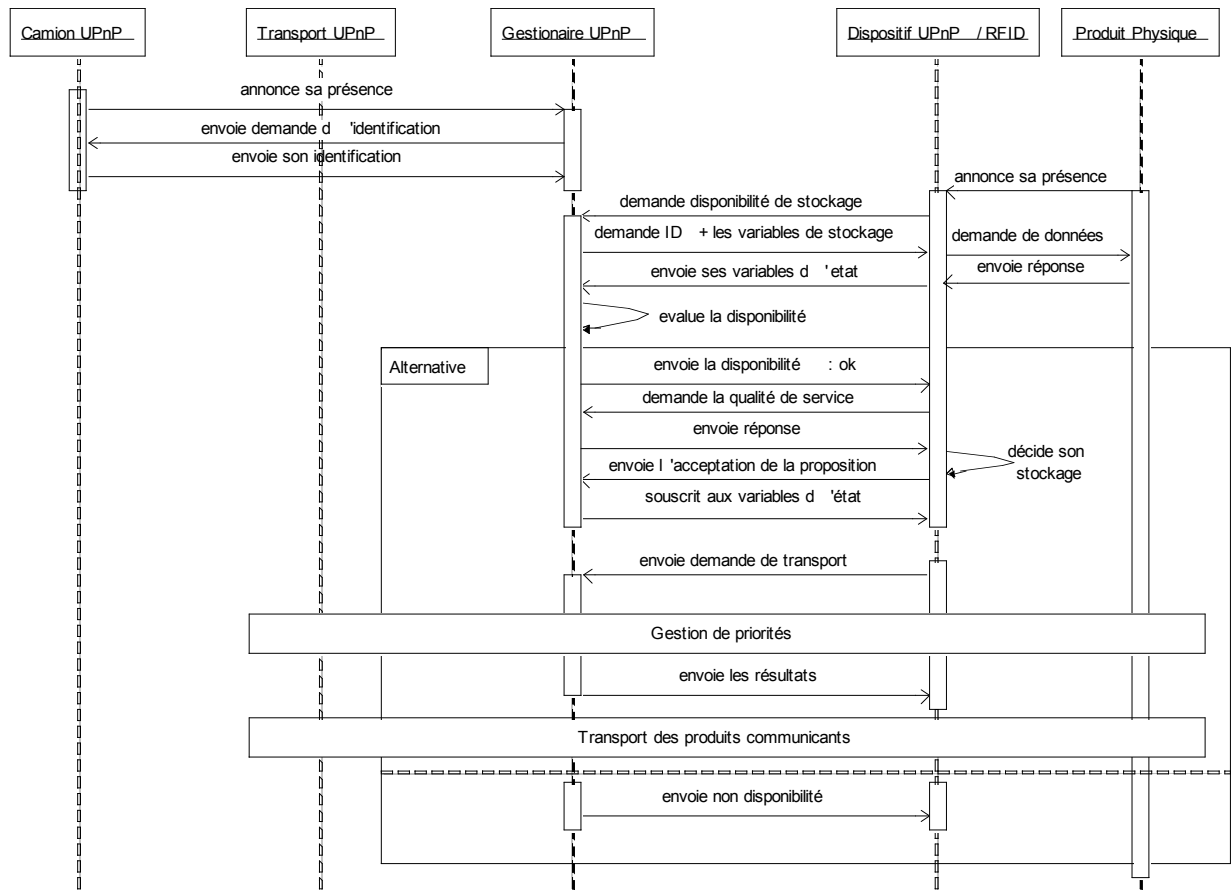


Figure 95 : Diagramme de séquence UML décrivant les interactions entre acteurs afin de stocker un objet physique dans l'entrepôt.

5.6.2 Transport des produits communicants dans l'entrepôt

La Figure 96 illustre les interactions entre les acteurs de l'entrepôt lorsqu'un objet physique est transporté par un chariot dans l'entrepôt. Dans ce cas, l'objet physique est un fournisseur de services. Au moment du rendez-vous entre le chariot et l'objet physique, le chariot cherche l'objet et lui demande sa localisation géographique et sa localisation fonctionnelle dans l'entrepôt. Ensuite, le chariot se déplace vers l'objet physique. Lorsque le chariot arrive à la localisation de l'objet, il vérifie l'état physique de l'objet, l'état opérationnel de l'objet et sa destination (le domaine fonctionnel suivant). Ensuite, le chariot prend l'objet physique et le transporte vers le domaine fonctionnel suivant. En même temps, il change l'état opérationnel de l'objet et actualise le domaine fonctionnel de l'objet (actuel, précédent et suivant). Une fois que l'objet arrive à sa destination, le chariot actualise sur l'objet communicant ses variables d'état. Chaque fois qu'une variable d'état est changée sur l'objet, le gestionnaire de l'entrepôt est informé parce qu'il est souscrit aux changements de l'objet.

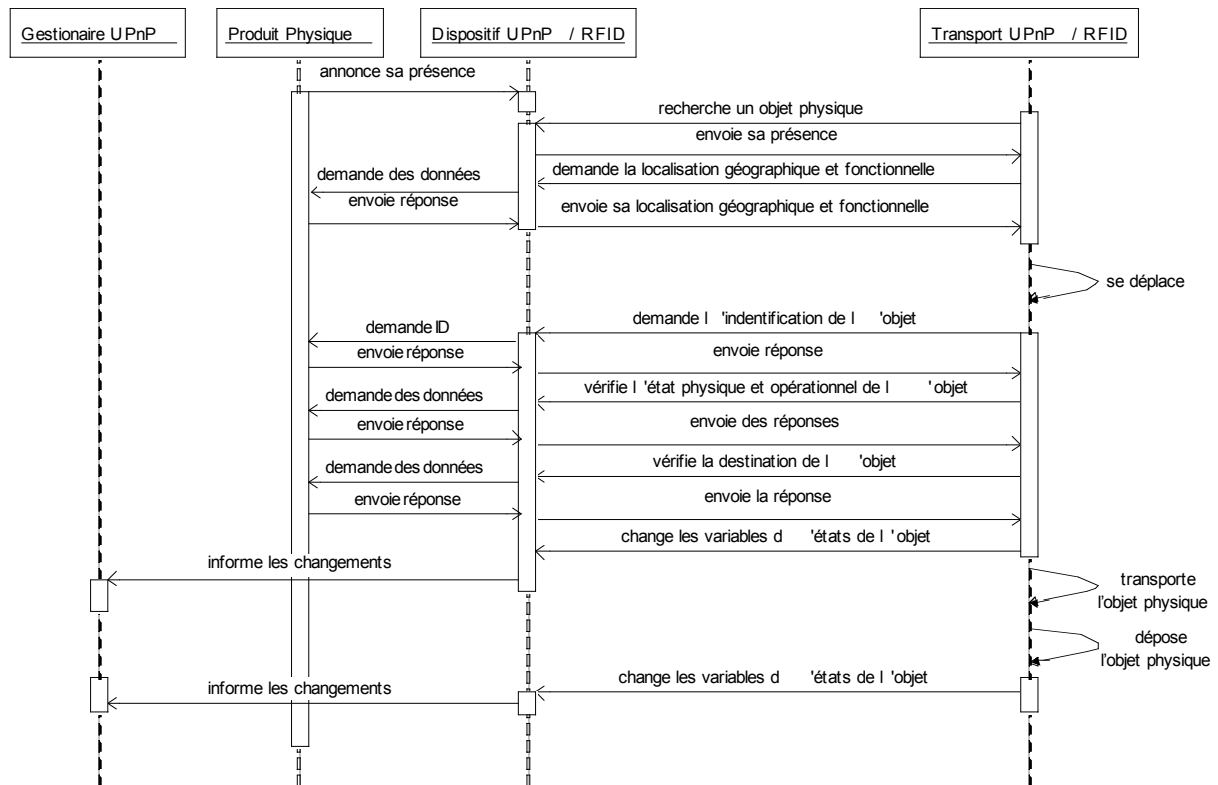


Figure 96 : Diagramme de séquence décrivant les interactions entre acteurs afin de transporter un objet physique dans l'entrepôt.

5.6.3 Gestion des alarmes dans l'entrepôt

La Figure 97 illustre les interactions entre un objet communicant actif et un capteur UPnP de l'entrepôt. L'objet communicant, afin de surveiller son environnement, fait une recherche pour identifier les capteurs disponibles dans le réseau UPnP. Ensuite, il demande la description des services offerts afin d'identifier les mieux adaptés à sa situation dans l'entrepôt. Puis, si l'objet est intéressé par les services proposés, il souscrit aux événements des capteurs. Ainsi, le capteur informe à l'objet communicant actif les changements dans les valeurs des variables souscrites.

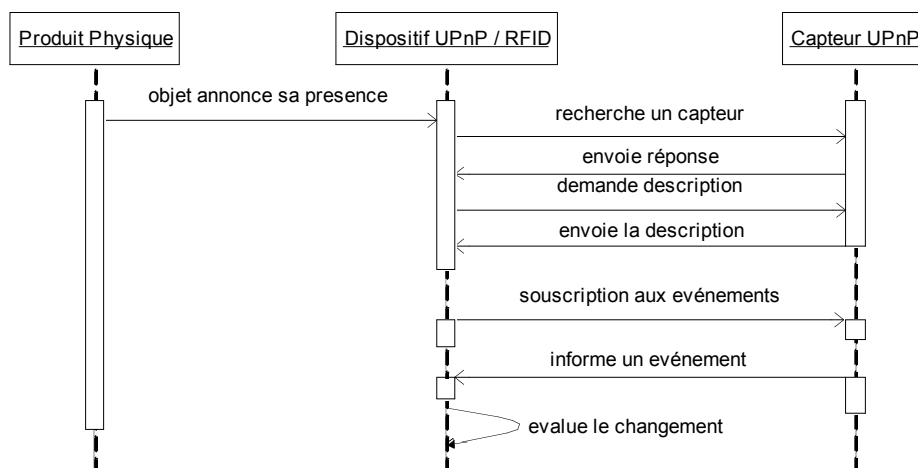


Figure 97 : Diagramme de séquence UML décrivant les interactions entre un objet communicant actif et un capteur dans l'entrepôt.

5.6.4 Sortie des produits communicants de l'entrepôt

La Figure 98 illustre les interactions entre les acteurs de l'entrepôt afin de gérer la sortie d'un objet physique. Lorsqu'un camion arrive à l'entrepôt pour transporter un produit vers une autre destination, il annonce son arrivée au gestionnaire de l'entrepôt. Le gestionnaire lui demande son identification et description afin de vérifier son identité et de lui permettre d'interagir avec les objets physiques. Ensuite, le camion recherche les objets physiques afin de les identifier et de leur demander de sortir de l'entrepôt. Lorsque l'objet physique reçoit l'indication de se diriger vers le camion, l'objet demande un transport en utilisant le mécanisme de gestion de priorités (Cf. Figure 91). Avec un chariot affecté, l'objet physique est transporté à la zone de sortie de l'entrepôt en suivant le modèle d'interactions décrit dans le point 5.6.2 (Cf. Figure 96).

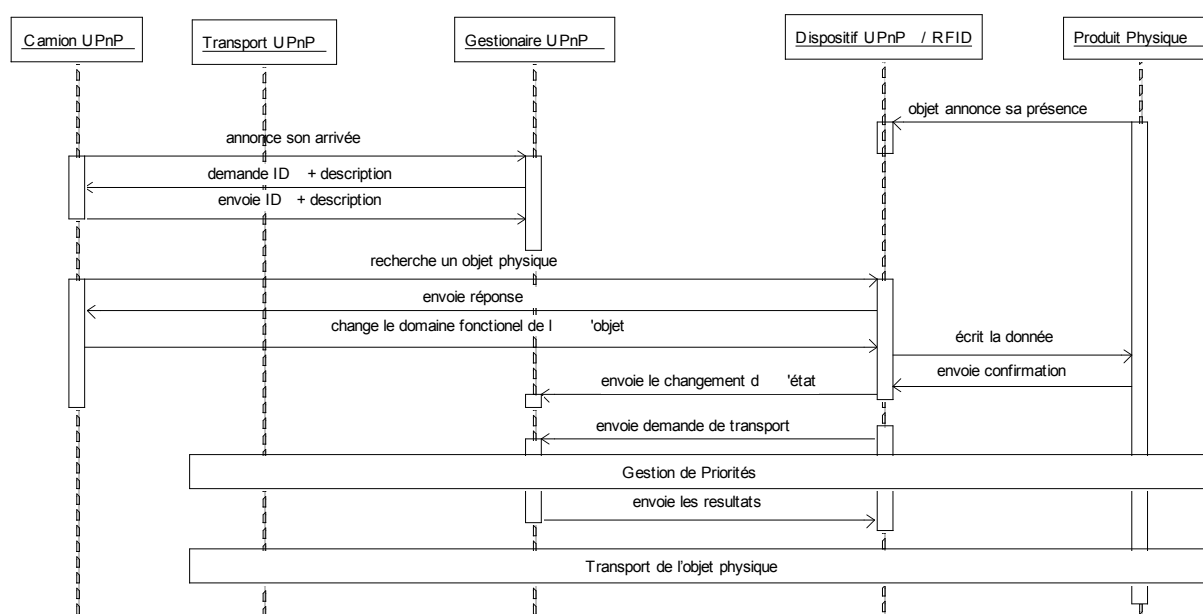


Figure 98 : Diagramme de séquence UML décrivant les interactions entre les acteurs afin de gérer la sortie des objets physiques de l'entrepôt.

5.7 Génération de démonstrateur de laboratoire

Dans cette partie, nous allons démontrer la faisabilité de notre approche sur la base d'un démonstrateur développé au laboratoire.

Le démonstrateur est constitué de trois parties dont la première représente un l'objet communicant passif / actif agissant comme un demandeur ou fournisseur de services. Afin d'implémenter le rôle de fournisseur de services dans l'objet communicant, nous avons intégré les classes de services définis auparavant grâce à l'utilisation des étiquettes électroniques attachées aux objets physiques. Ceci permet de répondre aux besoins d'interaction des acteurs de l'entrepôt. Afin d'implémenter le rôle de demandeur de services dans l'objet communicant actif nous allons intégrer des fonctionnalités dans le démonstrateur permettant à l'objet de demander des services vers une entité externe.

La deuxième partie représente un demandeur de services (point de contrôle UPnP) qui permettra l'interaction avec l'objet communicant passif à distance. Par rapport à notre cas d'étude, ce démonstrateur peut représenter les acteurs qui demandent des services à l'objet communicant tel que le gestionnaire de l'entrepôt ou un moyen de transport.

La troisième partie représente un capteur UPnP qui permettra à l'objet communicant actif de connaître le niveau de température et d'humidité dans son environnement.

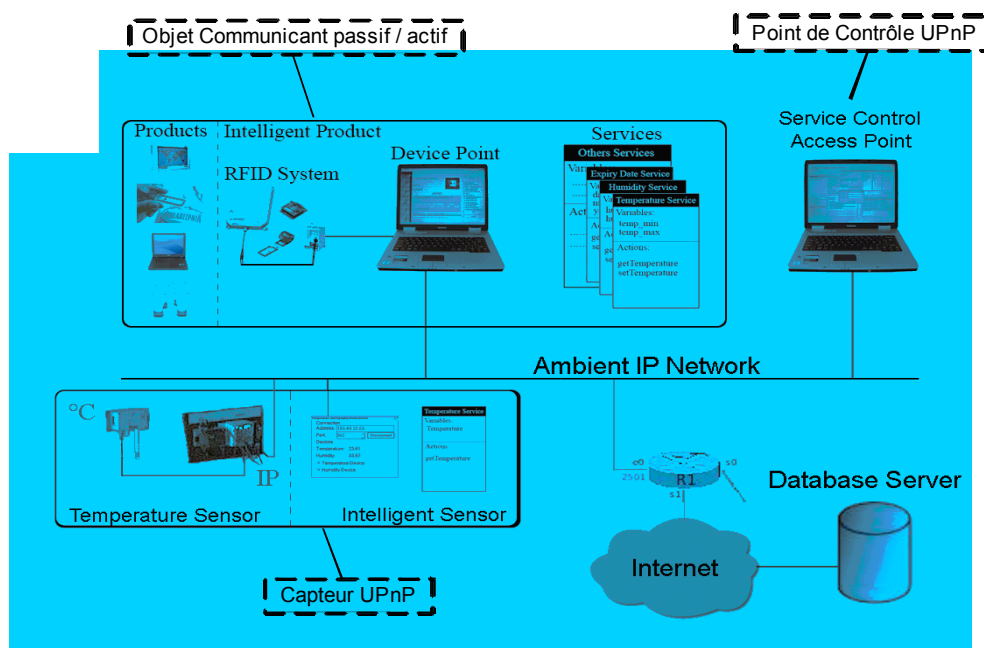


Figure 99 : Architecture de démonstration des interactions entre acteurs par les services.

La Figure 99 illustre l'architecture de démonstration que nous avons définie afin de tester les interactions dans l'architecture de services ambiants UPnP. Une base de données peut être accédée à distance pour obtenir l'information augmentée des objets physiques.

5.7.1 Démonstrateur objet communicant passif / actif

L'objet communicant passif est caractérisé pour offrir aux demandeurs de services un ensemble de services. Dans notre cas, les objets physiques offrent des services aux acteurs de l'entrepôt en s'appuyant sur les capacités de mémorisation fournies par des étiquettes électroniques RFID. Dans notre démonstrateur nous avons utilisé un système RFID OMRON V720S opérant avec des étiquettes électroniques I-CODE1 de 13.56 Mhz. La Figure 100 illustre la structure de la mémoire de cette étiquette électronique. L'étiquette contient 64 octets de mémoire. Les fonctions du système (SNR, protection d'écriture, EAS) sont composées de 5 pages de mémoire. Chaque page contient 4 octets (32 bits). Le FC (*Family Code*) et le AI (*Application ID*) sont utilisés pour enregistrer respectivement le type d'objet (produit, boîtes de produit, palette) et les services disponibles (actifs) d'un objet. L'aire utilisateur (*user area*) contient 11 pages (44 octets) de mémoire disponibles.

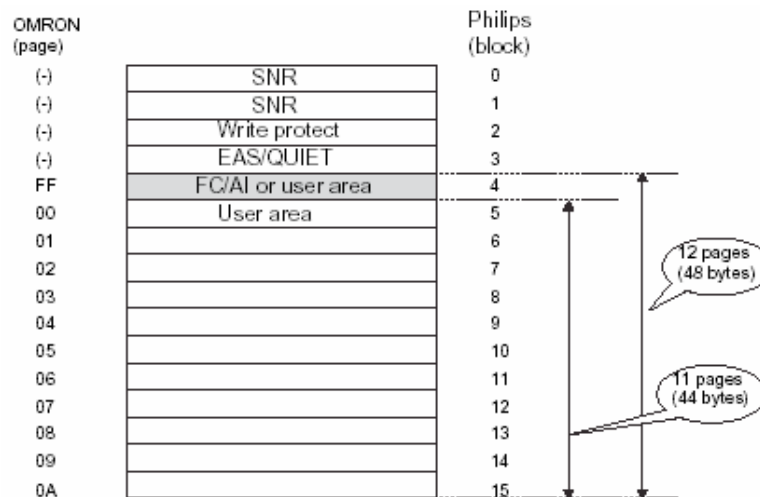


Figure 100 : Mémoire d'une étiquette électronique de produit communicant.

Dans notre démonstrateur, nous avons utilisé le mode de communication *Single Access Mode*, c'est-à-dire, le mode qui permet de lire ou écrire une étiquette à la fois. Cela veut dire que la relation objet physique - dispositif UPnP RFID est 1 : 1.

5.7.1.1 Création et description des services de l'objet communicant

Le processus de création des services d'un objet communicant est appuyé par l'outil *Service Author* fourni par Intel⁴⁵. Cet outil permet de générer et de valider automatiquement la description XML des services proposés pour un objet communicant. Pour créer un document XML, il faut définir les variables d'un service, et ensuite les actions qui vont être associées à ces variables. Une fois définies les variables et les actions dans l'outil *Service Author*, nous pouvons obtenir le fichier de description XML associé à un service. On va obtenir autant des fichiers de description XML que des services offerts par l'objet communicant. Ainsi, tous les services qui sont offerts par un objet communicant passif (décrits dans le point 5.3.3 de ce chapitre) peuvent être obtenus en utilisant cet outil.

Dans la Figure 101, nous illustrons un document XML décrivant le Service Composition d'une Boite. On observe deux actions permettant d'obtenir et de changer le nombre de produits contenus dans une boîte de produits. Les deux actions sont associées à la variable NrProduits. La variable **Code** sera utilisée pour représenter le résultat d'une action donnée. Si le résultat d'une action est correct et sans erreur, la valeur sera « 0 ». Par exemple quand un demandeur de services exécute une action donnée sur l'objet communicant afin d'écrire une donnée sur l'étiquette électronique, le Code informera que l'action a été correctement réalisée. Dans la partie inférieure de la figure, on observe la description des variables associées au service : type de variable et son caractère événementiel. La variable NrProduits est définie comme événementielle afin de pouvoir informer les intéressés d'un changement interne.

⁴⁵ www.intel.com/technology/upnp - Intel Tools for UPnP Technologies.

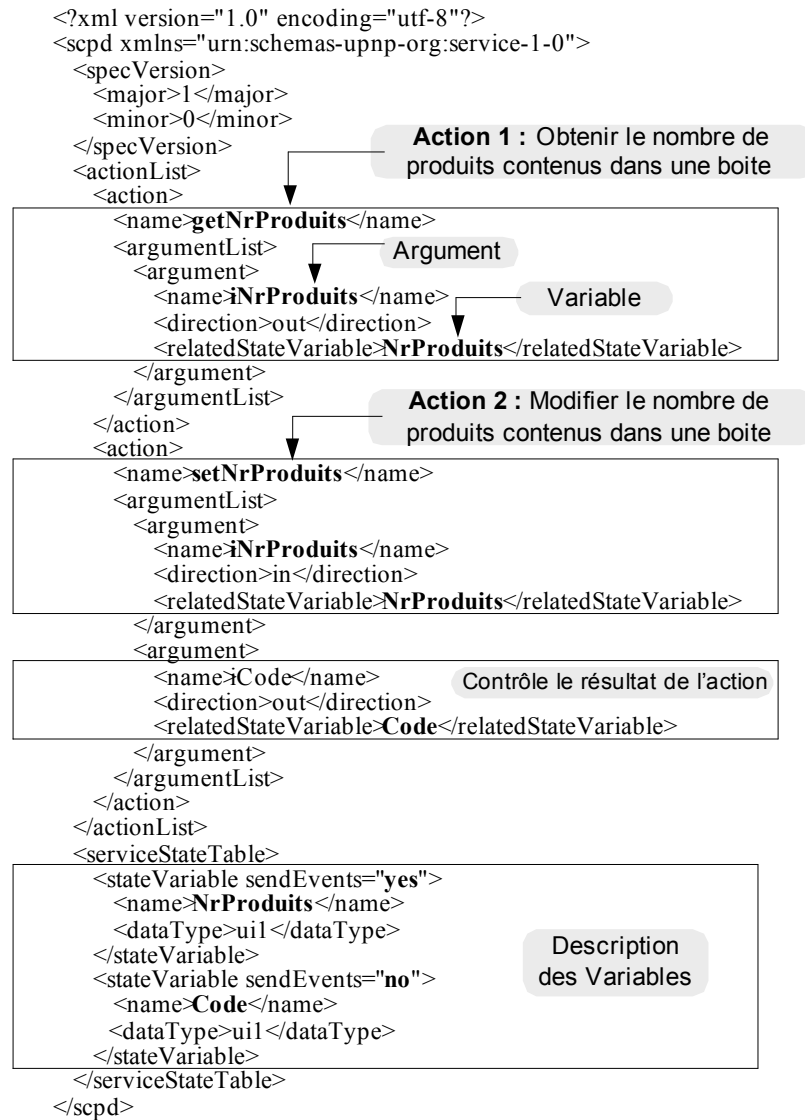


Figure 101 : Diagramme XML décrivant le service « Composition d'une boîte »

L'exemple de service expliqué sera associé à un objet physique « boîte de produits » et à un objet physique « palette ». Après avoir créé la description de tous les services d'un objet communicant, l'outil *Intel Device Spy (Universal Control Point)* permet de tester l'invocation d'action et la souscription aux événements des variables d'un objet communicant.

5.7.1.2 Génération de l'application logiciel

L'outil *Intel Device Builder* permet de générer automatiquement le code source de base pour créer un dispositif UPnP à partir d'un ensemble de services et leurs descriptions XML. Le code résultant incorpore les fonctionnalités UPnP et les méthodes permettant d'exécuter les actions du dispositif. L'*Intel Device Builder* travail sous le **Microsoft .Net Framework**. L'outil permet de choisir plusieurs plateformes de travail (Linux, Windows, Pocket PC,...). Nous avons travaillé avec la **plateforme .NET** avec le **langage de programmation C#**. L'environnement de travail **Visual Studio .NET 2005** permet d'éditer et de modifier le code de source de base. La compilation du code permet de créer l'application finale.

Dans notre cas, le code source doit être enrichi afin d'intégrer un système RFID dans le dispositif UPnP. La bibliothèque de communication fournie par *Franson Serial Tools*⁴⁶ nous a permis de gérer le port série de l'ordinateur afin de communiquer le système OMRON RFID depuis le dispositif UPnP. La communication entre le dispositif UPnP et le système RFID est réalisée au travers de fonctions supportées par le système OMRON RFID. Les fonctions les plus utilisées dans le démonstrateur sont : détection d'une étiquette, lecture d'une étiquette, écriture d'une étiquette, arrêt de la communication, test de communication. La Figure 102 illustre l'objet communicant obtenu.

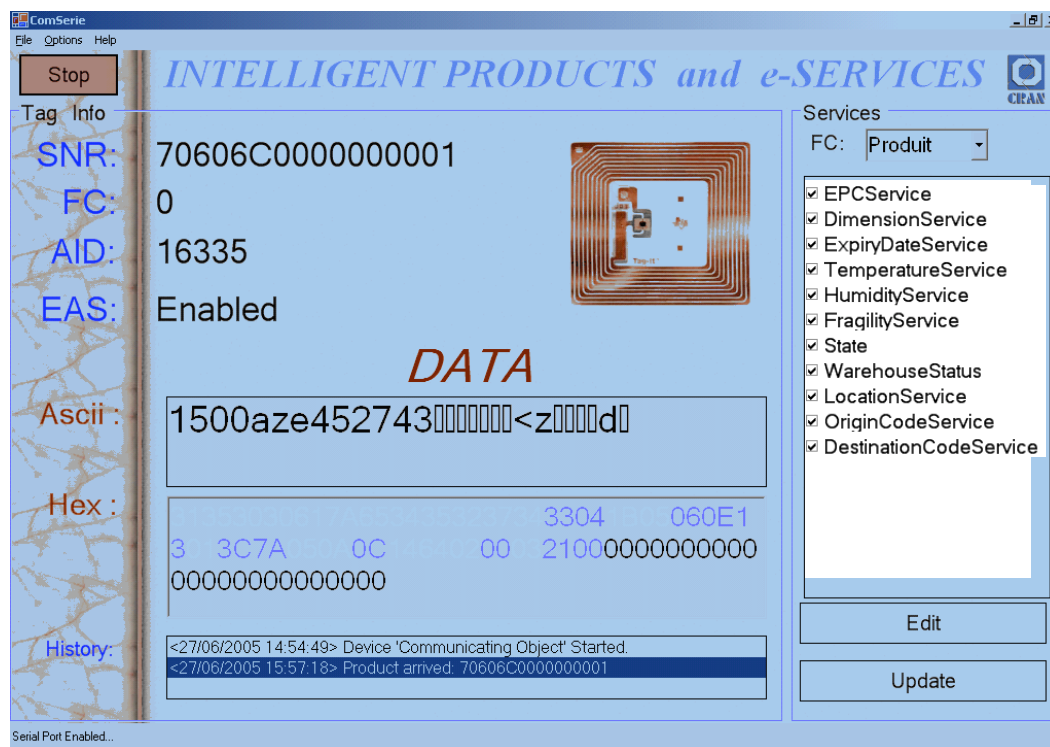


Figure 102 : Dispositif ou *Device* Point d'un Objet Communicant Passif / Actif.

Dans le côté gauche de la Figure 102 on observe les données qui vont permettre d'identifier l'étiquette électronique (SNR), le *Family Code* afin d'identifier le type d'objet (produit, boîte de produit, palette), l'AID (*Application ID*) afin d'identifier la liste de services actifs pour chaque objet physique, et le EAS (*Electronic Article Surveillance*) afin d'activer la fonctionnalité antivol de l'objet. Egalement, on observe les données stockées sur l'étiquette électronique (DATA) en format ASCII et Hexadécimal.

Dans la partie inférieure gauche de la figure, on présente l'historique des annonces de présence de l'objet physique dans la zone d'attraction RFID.

A droite dans la figure, on observe la liste de services actifs pour un type d'objet physique. La liste de chaque objet physique est obtenue grâce à l'Application ID. Dans le menu « *options* » (partie supérieure droite) il est possible de paramétrer l'usage du port série et du contrôleur RFID et d'ajouter de nouveaux services à la liste générique de services d'un objet physique.

Ce démonstrateur nous a permis de tester les interactions : produit communicant – point de contrôle / acteur de la chaîne logistique, produit communicant – capteur UPnP et produit communicant – base de

⁴⁶ <http://franson.com/serialtools/> - *Franson Serial Tools*

données (représentant un WMS). Dans ces interactions, l'objet communicant peut avoir un rôle de demandeur de services (vers des capteurs UPnP et vers une base de données) et de fournisseurs de services (vers un point de contrôle) dans un réseau ambiant UPnP.

5.7.2 Démonstrateur point de contrôle

Le démonstrateur point de contrôle permet de découvrir les objets communicants dans un réseau UPnP, d'exécuter des actions vers les objets communicants et de souscrire aux variables d'états d'un objet communicant. La création de ce démonstrateur a été appuyée par l'*Intel Device Builder* afin d'obtenir le code source de base en langage C#.

Dans la Figure 103, nous présentons l'application point de contrôle UPnP que nous avons développé. On observe à gauche la liste des dispositifs UPnP présents dans le réseau UPnP en incorporant la liste des objets communicant et des capteurs UPnP.

En bas à gauche, on observe la liste de services qu'offre l'objet communicant. A droite, on retrouve la description du service et la liste des actions associées à chaque service. Sur la liste des actions, la liste des variables associées au service est indiquée. Si la variable est événementielle, le point de contrôle permettra de souscrire à ses changements d'état. Au milieu de la figure, on observe la description du dispositif UPnP / RFID qui permet à un objet physique de devenir communicant dans un réseau UPnP.

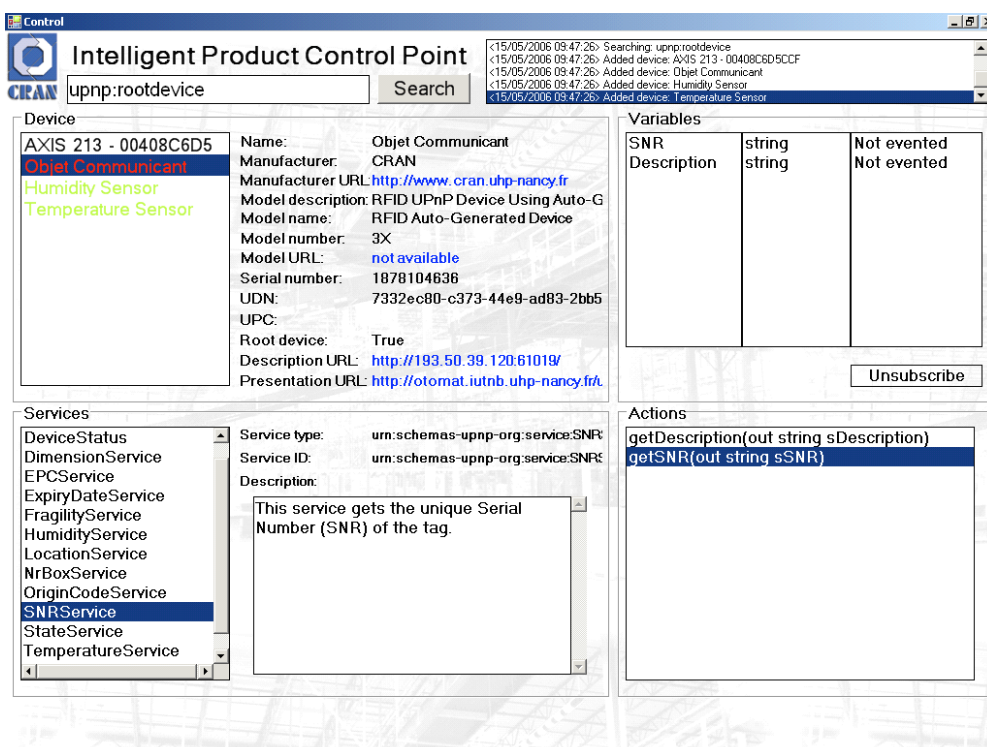


Figure 103 : Point de Contrôle UPnP qui permet d'interagir avec l'objet communicant.

Ce démonstrateur nous a permis d'interagir à distance avec l'objet communicant passif. Ce démonstrateur représente un acteur de l'entrepôt demandant des services à l'objet communicant. En particulier, la souscription aux événements d'un objet communicant a été testée avec ce démonstrateur. Lorsqu'un changement dans une variable d'état de l'objet communicant se produit, le démonstrateur « point de contrôle » UPnP est informé automatiquement.

5.7.3 Démonstrateur capteur UPnP

Dans la Figure 104 nous présentons l'application qui permet de transformer un capteur de température et d'humidité en un capteur UPnP.

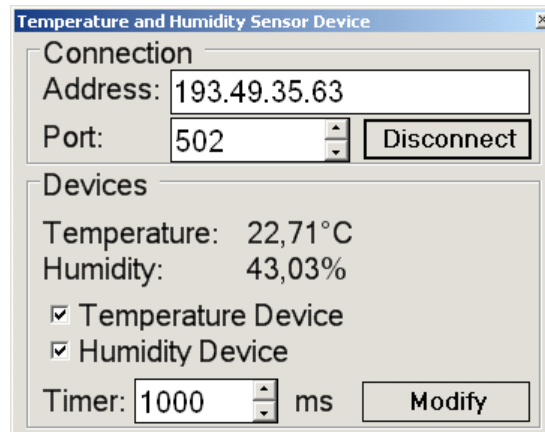


Figure 104 : Dispositif ou *Device Point* d'un capteur de température et humidité UPnP.

Dans notre cas, le démonstrateur Capteur UPnP nous a permis d'implémenter, dans l'objet communicant actif, la capacité à observer l'état de son environnement physique afin d'évaluer des actions face à d'éventuelles situations critiques d'entreposage.

5.8 Conclusion

Dans ce chapitre nous avons étudié l'application de notre méthodologie à la gestion d'un entrepôt en utilisant une approche d'objets / produits communicants. Comme résultat, nous avons proposé une architecture générique pour un entrepôt et nous avons défini des classes de services pour les acteurs de l'entrepôt. En plus, nous avons défini les relations entre les acteurs de l'entrepôt au travers des services que chaque entité peut demander aux autres acteurs du système dans une optique de services ambiants.

Nous avons détaillé le rôle d'un produit communicant, dans un scénario d'interactions ubiquistes, afin de montrer les potentialités de notre approche. La modélisation de services pour un produit communicant établit les interactions possibles entre acteurs. Ou vice-versa, en considérant les interactions souhaitées entre acteurs, nous pouvons définir les services, en suivant notre méthodologie, afin de faire interagir les acteurs d'un système par les services, que chacun met à disposition des autres acteurs, dans une infrastructure de services ambiants.

Egalement, nous avons présenté un démonstrateur de laboratoire afin de montrer la faisabilité d'implémentation de notre approche. Le démonstrateur « objet communicant » nous a permis de créer un objet communicant passif / actif dans un réseau UPnP. L'objet communicant créé est capable d'interagir avec un capteur UPnP dans le but de surveiller son environnement (interaction *Machine to Machine*). Ainsi, l'objet communicant peut réagir à des situations critiques, en envoyant, par exemple, un message d'alarme à un utilisateur proche.

Le démonstrateur « point de contrôle », dans un rôle d'acteur de l'entrepôt, effectivement peut demander des services à un objet communicant dans un réseau UPnP. Nous avons utilisé une base de

données UPnP pour simuler un WMS. Nous avons pu faire interagir l'objet communicant avec le « WMS » dans le but de permettre à l'objet d'évaluer son possible stockage dans un entrepôt et, ensuite, de prendre une décision.

6 Conclusions de la thèse

Nous avons étudié la problématique des produits communicants dans la chaîne logistique afin de leur permettre de devenir des acteurs dans leur cycle de vie. Cette hypothèse de travail a orienté le déroulement de cette thèse. Cette orientation nous a permis d'enrichir un produit physique avec de nouvelles capacités de mémorisation, de perception, de communication, d'action et de décision, afin de répondre aux besoins croissants de réactivité face aux clients et aux processus d'une chaîne logistique de plus en plus communicante, coordonnée, coopérative et compétitive. Ceci est justifié notamment par l'information demandée par de multiples acteurs aux produits physiques, en temps réel, afin de connaître ses états et son histoire quel que soit le moment ou le lieu. Particulièrement, un produit communicant, étant un acteur dans son cycle de vie, peut interagir avec d'autres acteurs de la chaîne logistique (des processus, des utilisateurs et d'autres produits) grâce à l'usage des technologies d'identification automatique et de communication ambiante. A cet égard, la notion ambiante permet aux produits d'interagir avec les acteurs au travers de services ambiants accessibles par des réseaux ambiants. Ces interactions ubiquistes confèrent au produit un rôle actif en pouvant demander et offrir des services aux acteurs de la chaîne logistique au long de sa trajectoire en passant par divers domaines de services, depuis sa fabrication jusqu'à son recyclage.

Plus précisément, en s'appuyant sur l'état de l'art des objets communicants, nous avons constaté l'existence de deux tendances pour fournir des services à la société des objets communicants. La première tendance est basée sur la miniaturisation des composants électroniques, de plus en plus puissants, capables d'intégrer effectivement des objets communicants dans une logique de dispositif intelligent (*smart device*). Cette tendance est principalement associée au domaine de télécommunications (téléphone portable, dispositif GPS, ...). La deuxième tendance est associée à une augmentation des capacités des objets de la vie quotidienne avec le support d'infrastructures informatiques afin de concevoir des objets communicants augmentés capables d'interagir avec des objets (communication *object to object* – O2O), avec des machines dans un esprit de communication machine to machine – M2M. Cette deuxième tendance a pour but d'étudier et tester de nouveaux domaines d'applications pour les *smarts objets*.

Dans le cadre de notre recherche, nous avons adopté une approche visualisant un produit physique comme un produit communicant capable d'agir comme un demandeur et fournisseur de services nécessitant une infrastructure de services ambiant pour opérer et devenir communicant dans son environnement. La condition initiale pour enrichir un produit physique exige son identification unique dans son domaine d'interaction. Cela permet d'initier une interaction personnalisée entre un produit et un acteur de la chaîne logistique. Dans nos recherches, nous avons déterminé que la technologie d'identification par radio fréquence est la plus adaptée et prometteuse dans le domaine de la chaîne logistique. Etant donné ceci, nous avons adopté la technologie RFID comme technologie de base pour identifier les produits physiques en permettant, de cette façon, l'union entre la représentation physique et virtuelle d'un produit. L'architecture de services ambiants qui satisfait pleinement notre objectif de recherche, est l'architecture de services ambiant UPnP (*Universal Plug and Play*).

Grâce à une analyse des caractéristiques nécessaires pour faire interagir un produit physique avec d'autres entités, nous avons proposé un modèle d'objet communicant. Ce modèle exige six fonctionnalités pour transformer un produit physique en produit communicant dans une approche de services ambiants : l'identification unique d'un produit, la recherche de services dans un réseau ambiant, la localisation logique et géographique d'un produit communicant, la description des services

du produit dans le réseau ambiant, l'invocation à distance des services d'un produit communicant et la gestion d'événements associés aux produits communicants.

La validation de notre approche de produits communicants a été appuyée par des tests d'intégration technologique entre la technologie RFID et l'architecture de services ambiants UPnP, et par la mise en place de services ambiants spécialisés pour supporter les interactions ubiquistes entre produits communicants, objets communicants (capteur) et système de contrôle (point d'interaction utilisateur) en utilisant des démonstrateurs de laboratoire.

Afin de guider la mise en place de notre approche, nous avons proposé une méthodologie pour concevoir des produits communicants sur la base d'extensions informationnelles des produits physiques, représentées par des services ambiants. Dans ce sens, la modélisation des services d'un produit communicant permet la conceptualisation des nouvelles capacités d'un produit communicant. La définition de ces services doit considérer l'information contextuelle d'un produit communicant. A ce propos, nous proposons de préciser la nature du produit communicant, ses domaines d'interaction (déterminés par les domaines de services fonctionnels et géographiques), et les profils des acteurs de la chaîne logistique.

Nous identifions deux perspectives dans cette ligne de recherche. Etant donné la standardisation internationale de l'approche ePCglobal permettant d'identifier un produit de façon unique, nous encourageons une standardisation des services pour les produits communicants. Notre approche méthodologique, s'appuie sur le langage XML pour permettre une description standardisée de fichiers de description de services. La définition de services standard requiert la participation de tous les membres de la chaîne logistique afin de permettre aux produits communicants de s'exprimer au long de leurs cycles de vie. Nous avons testé notre approche dans un réseau ambiant en laboratoire. L'interconnexion de réseaux ambiants nécessite des mécanismes de communication entre réseaux UPnP. Dans cet ordre d'idées, un test d'application dans toute la chaîne logistique avec des produits portant des services standardisés est une perspective de recherche très intéressante.

Comme deuxième perspective résultant de ces travaux, nous pouvons envisager l'impact du développement de produits communicants de plus en plus sensitifs, équipés de capteurs pour détecter, surveiller et contrôler l'environnement immédiat. Un produit communicant sensitif, équipé de capteurs, autonome avec sa propre source d'énergie et indépendant d'une architecture informatique de services, pouvant communiquer directement avec d'autres objets, des machines, permettrait de faire une réalité de la vision de Mark Weiser, dans un univers où le langage, les informations et les services sont à définir.

7 Références

- [Aarts et Appelo, 1999] Aarts E., Appelo L. Ambient intelligence: thuisomgevingen van de toekomst, IT Monitor 9/99, pp.7-11.
- [Agard et Tollenaere, 2002] Agard B, Tollenaere M. Conception d'assemblages pour la customisation de masse. Mécanique & Industries 3 :113–119, 2002.
- [Akyildiz et al., 2002] Akyildiz I.F., Su W., Sankarasubramaniam Y., Cayirci E. Wireless sensor networks: a survey. Computer Networks 38:393–422, 2002.
- [APICS, 1998] APICS Dictionary, American Production and Inventory Control Society. Cox J.F, Blackstone J.H. Jr, Publisher Apics, 9th Edition. USA.
- [APICS, 2001] APICS Dictionary, American Production and Inventory Control Society. Cox J.F, Blackstone J.H. Jr, Publisher Apics, 10th Edition. USA.
- [Ark et Selker, 1999] Ark W.S. Selker T. A look at human interaction with pervasive computers. IBM Systems Journal, Vol 38, N° 4, 1999.
- [Bajic et al., 1993] Bajic E., Chaxel F., Veron M. Intelligent parts management in a fully distributed architecture. IEPM (International conference on Industrial Engineering and Production Management) 1993. Mons (Belgique), 2-4 Juin 1993.
- [Bajic et al., 1999] Bajic E., Chaxel F., Richard J. Life cycle vehicle data management on holonic based concept. International Conference on Industrial Logistics, June 28 July 1, 1999, pp 121-129, St Petersburg, Russia.
- [Bajic et Cea, 2005] Bajic E., Cea A. Smart Objects and Services Modeling in the Supply Chain. 16th IFAC World Congress, Prague, Czech Republic, du 4 au 8 Juillet 2005.
- [Bajic et Chaxel, 2002a] Bajic E., Chaxel F. Holonic Manufacturing with Intelligent Objects. Fifth IFIP/IEEE International Conference on Information Technology for Balanced Automation Systems in Manufacturing and Services (BASYS'02), September 25-27, 2002, Cancun, Mexico.
- [Bajic et Chaxel, 2002b] Bajic E., Chaxel F. Auto-ID mobile information system for vehicle life cycle data management. Conference on Systems, Man and Cybernetics. Hammamet Tunisia 2002.
- [Bajic, 2004] Bajic E. Ambient Networking for intelligent objects management, mobility and services. Séminaire Institut For Manufacturing - IFM Université de Cambridge , Angleterre, 2 April 2004
- [Barralon et al., 2004] Barralon N., Nguyen V-T., Rey Gaëtan Rey. Techniques de couplage de bureaux : Ambient-Desktop comme illustration. 16ème Conférence Francophone sur l'Interaction Homme-Machine. Namur, Belgique 2004. ACM Press. Septembre 2004. pp. 61-68
- [Bauer et Odell, 2005] Bauer B., Odell J. UML 2.0 and agents: how to build agent-based systems with the new UML standard. Engineering Applications of Artificial Intelligence 18:141–157, 2005.
- [Beigl et al., 2001] Beigl M., Gellersen H-W, Schmidt A. Mediacups : experience with design and use of computer-augmented everyday artefacts. Computer Networks 35, 401-409, 2001
- [Beigl et al., 2004] Beigl M., Krohn A., Zimmer T. et Decker C. Typical Sensors needed in Ubiquitous and Pervasive Computing. First International Workshop on Networked Sensing Systems (INSS) 2004, Tokyo, Japan, June 22-23. 2004, pp 153-158
- [Beigl et Gellersen, 2003] Beigl M. Gellersen H. Smart-its: An embedded platform for smart objects. Proceedings of sOc'2003 (Smart Objects Conference), Grenoble, 2003.

- [Bilchev *et al.*, 2005] Bilchev G., Venousiou R., Foley J., Benyon P., Case S. Churcher G. Personal spaces – ePerSpace. *BT Technology Journal*. Vol 23 N° 3, July 2005.
- [Bohn *et al.*, 2005] Bohn J., Coroama V., Langheinrich M., Mattern F., Rohs M. Social, Economic, and Ethical Implications of Ambient Intelligence and Ubiquitous Computing. In: W. Weber, J. Rabaey, E. Aarts (Eds.): *Ambient Intelligence*, Springer-Verlag, pp. 5-29, 2005.
- [Bongaerts *et al.*, 2000] Bongaerts L., Monostori L., McFarlane D., Kádár B. Hierarchy in distributed shop floor control. *Computers in Industry* 43:123-137, 2000.
- [Borovoy *et al.*, 1996] Borovoy R., McDonald M., Martin F., Resnick M. Things that blink: Computationally augmented name tags. *IBM Systems Journal*, Vol 35, NOS 3&4, 1996.
- [Briscoe, 2004] Briscoe R. The implications of pervasive computing on network design. *BT Technology Journal*. Vol 22 N° 3 . July 2004.
- [Brock *et al.*, 2001] Brock D., Milne T., Kang Y. & Lewis B. The Physical Markup Language. Core Components: Time and Place. MIT Auto-ID Center. Technical Report. 2001.
- [Brock, 2001] Brock D. The Electronic Product Code (EPC) – A Naming Scheme For Physical Objects. MIT Auto-ID Center. Technical Report. 2001.
- [Casademont *et al.*, 2004] Casademont J., Lopez-Aguilera E., Paradells J., Rojas A., Calveras A., Barcelo F., Cotrina J. Wireless technology applied to GIS. *Computers & Geosciences* 30:671–682, 2004.
- [Cea *et al.*, 2004] Cea A., Bajic E., De Matteis P. Une Approche de Modélisation des Interactions Produits - Processus par les Objets Communicants. MOSIM2004: 5ème conférence francophone de MODélisation et SIMulation, Nantes, France, 1-3 septembre 2004.
- [Cea et Bajic, 2004] Cea A., Bajic E. Ambient Services for Smart Objects in the Supply Chain Based on RFID and UPnP Technology. 3rd MCPL'04, Third Conference on Management and Control of Production and Logistics, Santiago de Chile, Chile, 3 – 5 November 2004
- [Cea et Bajic, 2005] Cea A. Bajic E. Service Modeling for Intelligent Products in the Supply Chain. ICPR 18 - 18th International Conférence on Production Research - Fiasciano, Italy, 31/07-04/08 2005.
- [Château *et al.*, 2003] Château H., Girard D., Degueurce C., Denoix J-M. Analyse des contraintes méthodologiques liées à l'utilisation d'un système d'analyse cinématique tridimensionnelle fondé sur le principe de la triangulation ultrasonore. *ITBM-RBM* 24 :69–78, 2003.
- [Chaxel, 1995] Chaxel F. Contribution à la gestion et à la conduite des systèmes manufacturiers par les objets nomades de production. Doctorat de l'Université Henri Poincaré Nancy I, Spécialité Production Automatisée. 1999.
- [Cheverst *et al.*, 2002] Cheverst K., Mitchell K., Davies N. Exploring context-aware information push. *Personal and Ubiquitous Computing* 6:276-281, 2002
- [Chioiu *et al.*, 2003] Chioiu A., Barthel D., Le Menn P. Creating a Ubiquitous Environment with Smart-Tags. *Proceedings of sOc'2003 (Smart Objects Conference)*, Grenoble, 2003.
- [Ciarleta, 2002] Ciarletta Laurent. Contribution à l'évaluation des technologies de l'informatique ambiante. Thèse Doctorat Université Henri Poincaré – Nancy I. Laboratoire LORIA. 2002. France.
- [Ciminiera *et al.*, 2004] Ciminiera L., Maggi P., Sisto R. Implementing innovative services supporting user and terminal mobility: the SCARAB architecture. *The journal of systems and software* 72, 367-376, 2004.

-
- [Clarke et al., 2006] Clarke R., Twede D., Tazelaar J. Boyer K. Radio frequency identification (RFID) performance: the effect of tag orientation and package contents. *Packaging technology and science* 19: 45-54. 2006.
- [Console et al., 2003] Console L., Lombardi I., Gioria S. Personalized and adaptive services on board a car: an application for tourist information. *Journal of Intelligent Information Systems*, 21:3,249-284, 2003.
- [Coutaz et al., 2002] Coutaz J., Lachenal C., Bérard F., Barralon N. Quand les surfaces deviennent interactives. *Les cahiers du numérique 2002*, Lavoisier. Vol 3, Numéro 4. pp. 101-126
- [Coutaz et al., 2005] Coutaz J. Crowley J.L., Dobson S., Garlan D. Context is key. *Communications of the ACM*. March 2005. Vol. 48, N° 3.
- [Croom et al., 2000] Croom S, Romano P., Giannakis M. Supply chain management: an analytical framework for critical literature review. *European Journal of Purchasing & Supply Management* 6: 67-83, 2000.
- [Da Silveira et al., 2001] Da Silveira G., Borenstein D., Fogliatto F.S. Mass customization: Literature review and research directions. *Int. J. Production Economics* 72:1-13, 2001.
- [Dankwort et al., 2004] Dankwort C.W., Weidlich R., Guenther B., Blaurock J.E. Engineers' CAx education—it's not only CAD. *Computer-Aided Design* 36:1439–1450, 2004.
- [Desertot et al., 2005] Desertot M., Marin C., Donsez D. SensorBean : Un modélé à composants pour les services basés capteurs. *Journées Composants JC'05* April 2005, Le Croisic, France.
- [Dessenne, 2005] Dessenne Gérard-André. RFID : règlements et standards. *DATAcollection* Octobre 2005. pp. 24-32.
- [Dey, 2000] Dey A. Providing Architectural Support for Building Context-Aware Applications. PhD Thesis. Georgia Institute of Technology, November 2000.
- [Dey, 2001] Dey Anind K. Understanding and using context. *Personal and Ubiquitous Computing*. 2001. 5:4-7.
- [DOF-USA, 2001] DOF-USA. Global Positioning System Standard Positioning Service Performance Standard. Assistant Secretary of Defense for Command, Control, Communications, and Intelligence. October, 2001.
- [Duda, 2003] Duda A. Ambient Networking. *Proceedings of sOc'2003 (Smart Objects Conference)*, pp. 15-16, Grenoble, France, 2003.
- [Engels, 2003] Engels D. The Use of the Electronic Product Code™. MIT Auto-ID Center. Technical Report. 2003.
- [EURESCOM, 2000] EURESCOM. Smart Devices “When Things Start to Think”. Strategic Study. 2000.
- [Falk et al., 1999] Falk J., Redström, Björk. Handheld and ubiquitous computing. First International Symposium, HUC'99, Karlsruhe, Germany, September 1999. *Proceedings* Editors: H.W. Gellersen.
- [Ferber, 1995] Ferber Jacques. *Les systèmes multi-agents. Vers une intelligence collective*. InterEditions France. 1995. ISBN 2 7296 0665 3
- [Figge, 2004] Figge Stefan. Situation-dependent services - a challenge for mobile network operators. *Journal of Business Research* 57, 1416-1422, 2004.
- [FIPA, 2002] Foundation for Intelligent Physical Agents. FIPA Contract Net Interaction Protocol Specification. FIPA TC Communication. December 2002.

- [Finkenzeller, 2003] Finkenzeller K. RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification, Second Edition, 2003, John Wiley & Sons, Ltd. ISBN: 0-470-84402-7
- [Fishin et al., 2004] Fishin K., Roy S., Jiang B. Some Methods for Privacy in RFID Communication. Intel Corporation, IRS-TR-04-010, June 2004.
- [Flörkemeier et al., 2003] Flörkemeier C., Lampe M, Schoch T. The Smart Box Concept for Ubiquitous Computing Environments. Proceedings of sOc'2003 (Smart Objects Conference), pp. 118-121, Grenoble, 2003.
- [Fowler, 2004] Fowler M. UML Distilled. Third Edition. Addison-Wesley. 2004. ISBN 0 321 19368 7
- [France Télécom, 2006] France Télécom, FING, Syntec Informatique. Livre blanc. Machine to machine. Enjeux et perspectives. Mars 2006.
- [Fuatealba, 2005] Fuatealba Cecilia. Approche biométrique pour l'identification automatique des produits bois. Thèse présentée pour l'obtention du Docteur de l'université Henri Poincaré, Nancy 1. Discipline : Sciences du bois, Novembre 2005.
- [Fujinami et al., 2005] Fujinami K., Murata K., Nakajima T. Augmentation of everyday artefacts for context-aware application's building blocks. Proceedings of the International Workshop on Smart Objects Systems. UbiComp 2005. 69-76, Tokyo, Japan. 2005.
- [Geib et al., 1999] Geib J-M, Gransart C., Merle Ph. Corba. Des concepts à la pratique. Laboratoire d'informatique fondamentale de Lille. 2e édition. Dunod. ISBN 2 10 004806 6.
- [Georgakopoulos et al., 2002] Georgakopoulos D., Schuster H., Cichocki A., Baker D. Process-based e-service composition for modelling and automating zero latency supply chains. Information systems frontiers 4:1, 33-54, 2002.
- [Gershenfeld, 1999] Gershenfeld N. When things start to think. Henry Holt & Company; 1st edition. USA. 1999.
- [Ghorbel et al., 2004] Ghorbel M., Segarra M.T, Kerdreux J., Keryell R., Thépaut A., Mokhtari M. Networking and Communication in the Smart Home for People with Disabilities. 9th International Conference on Computers Helping People with Special Needs, Paris, juillet 2004.
- [Govea et Barbeau, 2000] Govea J. and Barbeau M. Comparison of Bandwidth Usage: Service Location Protocol and Jini. Technical Report TR-00-06, School of Computer Science, Carleton University, September 2000.
- [Gransart et al., 2001] Gransart C., Rolland P.A., Simplot D. Liaison de Proximité Haut Débit entre Objets Mobiles Communicants. Séminaire sur les objets communicants 2001, Meylan, France, 2001.
- [Gu et al., 2005] Gu T., Keng H., Qing D. A service-oriented middleware for building context-aware services. Journal of Network and Computer Applications 28, 1-18, 2005.
- [Holmquist et al., 2003] Holmquist L.E., Mazé R., Ljungblad S. Designing tomorrow's smart products – experience with the smart-its platform. Proceedings of Designing User Experience (DUX), 2003.
- [Huang et Garcia-Molina, 2004] Huan Y., Garcia-Molina H., Publish/Subscribe in a Mobile Environment. Wireless Networks 10, 643-652, 2004. Kluwer Academic Publishers.
- [Ishii et Ullmer, 1997] Ishii H., Ullmer B. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. In Proceedings of CHI'97, March 22-27; 1997, ACM

-
- [ISO 11784, 1996] ISO 11784:1996. Radio frequency identification of animals -- Code structure. 1996
- [ISO 11785, 1996] ISO 11785:1996. Radio frequency identification of animals -- Technical concept. 1996
- [ISO 14223-1, 2003] ISO 14223-1:2003. Radiofrequency identification of animals -- Advanced transponders -- Part 1: Air interface. 2003
- [ISO 15961, 2004] ISO/IEC 15961:2004. Information technology -- Radio frequency identification (RFID) for item management -- Data protocol: application interface. 2004
- [ISO 15962, 2004] ISO/IEC 15962:2004. Information technology -- Radio frequency identification (RFID) for item management -- Data protocol: data encoding rules and logical memory functions. 2004.
- [ISO 15963, 2004] ISO/IEC 15963:2004. Information technology -- Radio frequency identification for item management -- Unique identification for RF tags. 2004
- [ISO 18000-1, 2004] ISO/IEC 18000-1. Information technology -- Radio frequency identification for item management -- Part 1: Reference architecture and definition of parameters to be standardized. 2004
- [ISO 18000-2, 2004] ISO/IEC 18000-2:2004. Information technology -- Radio frequency identification for item management -- Part 2: Parameters for air interface communications below 135 kHz. 2004
- [ISO 18000-3, 2004] ISO/IEC 18000-3:2004. Information technology -- Radio frequency identification for item management -- Part 3: Parameters for air interface communications at 13,56 MHz. 2004
- [ISO 18000-4, 2004] ISO/IEC 18000-4:2004. Information technology -- Radio frequency identification for item management -- Part 4: Parameters for air interface communications at 2,45 GHz. 2004
- [ISO 18000-6, 2004] ISO/IEC 18000-6:2004. Information technology -- Radio frequency identification for item management -- Part 6: Parameters for air interface communications at 860 MHz to 960 MHz. 2004
- [ISO 18000-7, 2004] ISO/IEC 18000-7:2004. Information technology -- Radio frequency identification for item management -- Part 7: Parameters for active air interface communications at 433 MHz. 2004
- [ISO 18046, 2005] ISO/IEC TR 18046:2005. Information technology -- Automatic identification and data capture techniques -- Radio frequency identification device performance test methods. 2005
- [ISO 18047-3, 2004] ISO/IEC TR 18047-3:2004. Information technology -- Radio frequency identification device conformance test methods -- Part 3: Test methods for air interface communications at 13,56 MHz. 2004
- [ISO 18047-4, 2004] ISO/IEC TR 18047-4:2004. Information technology -- Radio frequency identification device conformance test methods -- Part 4: Test methods for air interface communications at 2,45 GHz. 2004
- [ISO 18047-7, 2005] ISO/IEC TR 18047-7:2005. Information technology -- Radio frequency identification device conformance test methods -- Part 7: Test methods for active air interface communications at 433 MHz. 2005

- [ISO 19762-1, 2005] ISO/IEC 19762-1:2005. Information technology -- Automatic identification and data capture (AIDC) techniques -- Harmonized vocabulary -- Part 1: General terms relating to AIDC. 2005
- [ISO 19762-2, 2005] ISO/IEC 19762-2:2005. Information technology -- Automatic identification and data capture (AIDC) techniques -- Harmonized vocabulary -- Part 2: Optically readable media (ORM). 2005
- [ISO 19762-3, 2005] ISO/IEC 19762-3:2005. Information technology -- Automatic identification and data capture (AIDC) techniques -- Harmonized vocabulary -- Part 3: Radio frequency identification (RFID). 2005.
- [ISO 8402, 1994] ISO 8402:1994. Quality management and quality assurance – Vocabulary. Edition 2.
- [ISO OSI, 1994] ISO/IEC 7498-1:1994 Ed. 2. Information technology -- Open Systems Interconnection -- Basic Reference Model: The Basic Model. 1994.
- [ITEA Ambience, 2004] ITEA Ambience. Project Result Context-Aware Environments for Ambient Services. October 2004.
- [Jacquemin et al., 2005] Jacquemin C., Afonso A., Blum A., de Laubier S., Denis M., Folch H., Genevois H., Katz B., Nugier S., and Schnell N. (2005). Design d'environnements multimodaux interactifs communicants. In Proceedings, Hypermedias Hypertexts, Products, Tools and Methods H2PTM'05, Paris, France.
- [Jammes et Smit, 2005] Jammes F., Smit H. Service-oriented paradigms in industrial automation. Proceedings of the 23rd IASTED - International Multi-Conference Parallel and Distributed Computing and Networks. February 15-17, Austria, 2005.
- [Jansen et Krabs, 1999] Jansen R., Krabs A. Automatic identification in packaging – Radio frequency identification in multiway systems. Packaging technology and science 12, pp 229-234, 1999.
- [Jansen-Vullers et al., 2003] Jansen-Vullers M.H., Van Dorp C.A., Beulensb A.J.M. Managing traceability information in manufacture. International Journal of Information Management 23:395–413, 2003.
- [Jarvis et al., 2006] Jarvis J., Rönnquist, McFarlane D., Jain L. A team-based holonic approach to robotic assembly cell control. Journal of Network and Computer Applications 29: 160-176, 2006.
- [Jeronimo et Weast, 2003] Jeronimo M. and Weast J. UPnP Design by Example. INTEL Press. Usa.
- [Jia et al., 2004] Jia S., Hada Y., Takase K. Distributed Telerobotics system based on Common Object Request Broker Architecture. Journal of Intelligent and Robotics Systems 39: 89-103, 2004.
- [Jiao et Tseng, 1999] Jiao J, Tseng M. A methodology of developing product family architecture for mass customization. Journal of Intelligent Manufacturing: 10,3-20, 1999.
- [Kadima et Monfort, 2003] Kadima H., Monfort Valérie. Les Web Services. Dunod, Paris, 2003 ISBN 2 10 006558 0.
- [Kärkkäinen et al., 2003a] Kärkkäinen M., Holmström J., Främling K., Arto K. Intelligent products – a step towards a more effective project delivery chain. Computer in Industry 50, pp.141-151. 2003
- [Kärkkäinen et al., 2003b] Kärkkäinen M., Ala-Risku T., Främling K. The product centric approach: a solution to supply network information management problems?. Computers in Industry 52, pp. 147-159. 2003.

-
- [Kawsar et al., 2005] Kawsar F., Fujinami K. and Nakajima T. Augmenting Everyday Life with Sentient Artefacts. Joint sOc-EUSAI Conference (Smart Objects Conference - European Symposium on Ambient Intelligence), Grenoble, France, pp. 141-146, 2005.
- [Keskilammi et al., 2003] Keskilammi M., Sydänheimo L., Kivikoski M. Radio Frequency technology for automated manufacturing and logistic control. Part 1: Passive RFID Systems and the effects of antenna parameters on operational distance. *International Journal of Advanced Manufacturing Technology* 21:769-774. 2003
- [Kintzig et al., 2002] Kintzig C., Poulain G., Privat G., Favennec P. *Objets Communicants*. Hermes Science. France Télécom R&D et Lavoisier, Paris, 2002.
- [Kintzig et al., 2003] Kintzig C., Poulain G., Privat G. and Favennec P. (eds). *Communicating with Smart Objects: Developing Technology for Usable Pervasive Computing Systems*. Kogan Page Science. 2003.
- [Kiritsis et al., 2003] Kiritsis D., Bufardi A., Xirouchakis P. Research issues on product lifecycle management and information tracking using smart embedded systems. *Advanced Engineering Informatics* 17:189–202, 2003.
- [Knospe et Pohl, 2004] Knospe H., Pohl H. RFID Security. Information Security Technical Report. Vol. 9, N° 4, 2004. Elsevier Ltda.
- [Koestler, 1967] Koestler A., *The ghost in the machine*, ISBN 0-14-019162-5.
- [Kovács et al., 2006] Kovács G., Kopácsi S., Haidegger G., Michelini R. Ambient Intelligence in Product Life-cycle Management. *Engineering Applications of Artificial Intelligence*. Accepted 26 January 2006. In Press.
- [Lahlou et al., 2005] Lahlou S., Langheinrich M., Röcker C. Privacy and Trust Issues with Invisible Computers. *Communications of the ACM*. March 2005. Vol. 48, N° 3.
- [Lamothe, 2006] Lamothe J., Hadj-Hamou K., Aldanondo M. An optimization model for selecting a product family and designing its supply chain. *European Journal of Operational Research* 169:1030–1047. 2006.
- [Lampe et Flörkemeier, 2004] Lampe M., Flörkemeier C. The Smart Box Application Model. In: Alois Ferscha, Horst Hoertner, Gabriele Kotsis (Eds.): *Advances in Pervasive Computing*, Austrian Computer Society (OCG), Vienna, Austria, April 2004.
- [Langheinrich et al., 2000] Langheinrich M., Mattern F., Römer K., Vogt H. First Steps Towards an Event-Based Infrastructure for Smart Things. *Ubiquitous Computing Workshop (PACT 2000)*, Philadelphia, PA., October 15-19, 2000.
- [Lejeune et al., 2005] Lejeune M.A., Yakova N. On characterizing the 4 C's in supply chain management. *Journal of Operations Management* 23: 81–100, 2005.
- [Leymann et Roller, 2005] Leymann F., Roller D. Modeling business processes with BPEL4WS. *Information Systems and E-Business Management (ISeB) 2005*. ISSN: 1617-9846 (Paper) 1617-9854 (Online).
- [Lindwer et al., 2003] Lindwer M., Marculescu D., Basten T., Zimmermann R., Marculescu R., Jung S., Cantatore E. Ambient Intelligence Visions and Achievements: Linking Abstract Ideas to Real-World Concepts. *Proc. Design, Automation and Test in Europe Conference. (DATE 2003)*, Munich, Germany, March 2003 (Supported by IEEE)
- [Loke et al., 2005] Loke S.W., Krishnaswamy S., Naing T.T. Service Domains for Ambient Services: Concept and Experimentation. *Mobile Networks and Applications* 10, 395-404, 2005.

- [Marin-Perianu et al., 2005] Marin-Perianu R., Hartel P., Scholten H. A Classification of Service Discovery Protocols. Centre for Telematics and Information Technology, Univ. of Twente, The Netherlands, Technical report nr. TR-CTIT-05-25, Jun. 2005.
- [Mark, 1999] Mark W. Turning pervasive computing into mediated spaces. IBM Systems Journal, Vol 38, N° 4, 1999.
- [Mason et al., 2003] Mason S.J., Ribera P.M., Farris J.A., Kirk R.G. Integrating the warehousing and transportation functions of the supply chain. Transportation Research Part E 39, 141–159, 2003.
- [Mathieu et al., 2001] Mathieu O, Montgomery D., Rose S. Empirical Measurements of Service Discovery Technologies. Pervasive Computing 2001, May 2001, Maryland.
- [Mattern et Sturm, 2003] Mattern F., Sturm P. From Distributed Systems to Ubiquitous Computing – The State of the Art, Trends, and Prospects of Future Networked Systems. In: Klaus Irmscher, Klaus-Peter Fähnrich (Eds.): Proc. KIVS 2003, pp. 3-25, Springer-Verlag, 2003.
- [Mattern, 2003] Mattern F. From Smart Devices to Smart Everyday Objects (Extended Abstract). Proceedings of sOc'2003 (Smart Objects Conference), pp. 15-16, Grenoble, France, 2003.
- [Mattern, 2005] Mattern F. Ubiquitous Computing: Scenarios from an informatized world. In: Axel Zerdick, Arnold Picot, Klaus Schrape, Jean-Claude Burgelman, Roger Silverstone, Valerie Feldmann, Christian Wernick, Carolin Wolff (Eds.): E-Merging Media - Communication and the Media Economy of the Future, Springer-Verlag, pp. 145-163, 2005.
- [Maxwell et Van der Vorst, 2003] Maxwell D., Van der Vorst R. Developing sustainable products and service. Journal of Cleaner Production 11:883–895.2003
- [McFarlane et al., 2003] McFarlane D., Sarma S., Chirn Jin Lung, Wong C.Y., Ashton K. Auto ID systems and intelligent manufacturing control. Engineering Applications of Artificial Intelligence. 16 (2003) 365-376.
- [McFarlane, 2002] McFarlane D. Auto ID based control systems – An overview. IEEE International Conference on Systems, Man and Cybernetics. Hammamet Tunisia 2002.
- [McFarlane et al., 2002] McFarlane D., Sarma S., Chirn Jin Lung, Wong C.Y., Ashton K.. The intelligent product in manufacturing control and management. 15th Triennial World Congress, Barcelona, Spain. 2002
- [Melchor, 2004] Melchor Miguel. Jini Evaluation Report. Service Infrastructure for Real-time Embedded Networked Applications. ITEA (Information Technology for European Advancement) 02014 Project. 2004.
- [Metras, 2005] Metras H. RFID tags for ambient intelligence: present solutions and future challenges. Joint sOc-EUSAI Conference (Smart Objects Conference - European Symposium on Ambient Intelligence) , Grenoble, France, pp. 43-46, 2005.
- [Metro Group, 2004] Metro Group Future Store Initiative. RFID: Une innovation prometteuse. Intégration de la technologie RFID à la chaîne de valeur. SAP France - Intel. Metro AG 2004.
- [MIT Project Oxygen, 2002] MIT Project Oxygen. Pervasive, Human - centered computing. MIT Project Oxygen. MIT Laboratory For Computer Science. MIT Artificial Intelligence Laboratory. Second Printing 2002.
- [Mokhtari et al., 2005] Mokhtari M., Ghorbel M., Kadouche R. Mobilité et Services : Application aux aides technologiques pour les personnes handicapées. 7th International Symposium on Programming and Systems (ISPS) Algiers May 2005.

-
- [Morel *et al.*, 2003] Morel G., Panetto H., Zaremba M., Mayer F. Manufacturing Enterprise Control and Management System Engineering: paradigms and open issues. *Annual Reviews in Control* 27, 199–209, 2003.
- [Morgenroth *et al.*] Morgenroth D., Hales T., Fobes K. Another link in the chain. *Card technology Today*, Volume 16, Issue 4, April 2004.
- [Naing *et al.*, 2003] Naing T.T, Loke S.W., Krishnaswamy S. A service-domain based approach to computing ambient services. Forum Session at the First International Conference on Service Oriented Computing (ICSOC03), Technical Report November 2003, Italy.
- [Noore *et al.*, 2004] Noore A., Tungala N., Houck M. Embedding biometric identifiers in 2D barcodes for improved security. *Computer & security* 23, 679-686, 2004.
- [Noury *et al.*, 2003] Noury N., Virone G., Ye J., Rialle V, Demongeot J. Nouvelles directions en habitats intelligents pour la santé. *ITBM-RBM* 24, 122-135, 2003.
- [O’Sullivan, 2002] O’Sullivan Justin. What’s in a Service? Towards accurate description of non-functional service properties. *Distributed and Parallel Databases*, 12, 117-133, 2002.
- [OASIS SAML, 2004] OASIS Security Services (SAML) TC. <http://www.oasis-open.org/committees/security>
- [OASIS WSBPEL, 2003] OASIS Web Services Business Process Execution Language (WSBPEL) TC. <http://www.oasis-open.org/committees/wsbpel>
- [OASIS WSS, 2003] OASIS Web Services Security (WSS) TC. <http://www.oasis-open.org/committees/wss>
- [OASIS XACML, 2001] OASIS eXtensible Access Control Markup Language (XACML) TC. <http://www.oasis-open.org/committees/xacml>
- [Oat Systems et MIT Auto-ID Center, 2002] Oat Systems & MIT Auto-ID Center. The Object Name Service. Technical manual. 2002
- [Omae *et al.*, 2001] Omae M., Fujioka T., Miyake K. Relative position measurement of neighboring vehicles using DGPS and inter-vehicle communication. *JSAE Review* 22:75-80, 2001.
- [OMG, 2004] OMG. Common Object Request Broker Architecture: Core Specification. Version 3.0.3. 2004.
- [Omron, 2001] Omron Corporation. V720-series Electromagnetic Inductive RFID System. Operation Manual. August 2001.
- [Oprescu, 2004] Oprescu M. Justinian. Découverte et composition de services dans des réseaux ambiants. Thèse Docteur de l’INPG Spécialité Informatique : Systèmes et Communications. 2004. France.
- [OSGi, 2004] OSGi Alliance. About the OSGi Service Platform. Technical Whitepaper. Revision 3.0. 2004.
- [Panozzo *et al.*, 1999] G. Panozzo G., Minotto G., Barizza A. Transport et distribution de produits alimentaires: situation actuelle et tendances futures. *International Journal of Refrigeration*. 22 : 625-639, 1999.
- [Paret, 2001] Paret D. Identification radiofréquence et cartes à puce sans contact. Dunod, Paris, 2001. ISBN 2 10 005955 6
- [Paret, 2003] Paret D. Applications en identification radiofréquence et cartes à puce sans contact. Dunod, Paris, 2003. ISBN 2 10 005778 2

- [Parlikad et McFarlane, 2004] Parlikad A.K., McFarlane D. Investigating the role of product information in end-of-life decisions making. INCOM 2004: 11th IFAC Symposium on Information Control Problems in Manufacturing. Salvador da Bahia, Brazil. 2004.
- [Penttilä et al., 2005] Penttilä K., Keskilammi M., Sydänheimo, Kivikoski M. Radio frequency technology for automated manufacturing and logistics control. Part 2: RFID antenna utilisation in industrial applications. International Journal of Advanced Manufacturing Technology. Accepted : 10 may 2005.
- [Philips HomeLab, 2003] Philips HomeLab. 365 days' Ambient Intelligent research in HomeLab. Philips Research. April 2003.
- [Praca et Barral, 2001] Praca D., Barral C. From smart cards to smart objects: the road to new smart technologies. Computer Networks 36, 381-389, 2001.
- [Privat, 2000] Privat G. A system-architecture viewpoint on smart networked devices. Microelectronic Engineering 54, 193-197, 2000.
- [Rantanen et al., 2002] Rantanen J., Impiö J., Karinsalo T., Malmivaara M., Reho A., Tasanen M., Vanhala. Smart Clothing Prototype for the arctic environment. Personal and Ubiquitous Computing 6, 3-16, 2002.
- [Reeves et Nass, 1996] Reeves B. et Nass C. The Media Equation. How People Treat Computers, Television, and Media Like Real People and Places. Cambridge University Press, 1996.
- [Rheingold, 2005] Rheingold H. Foules intelligentes. La révolution qui commence. Traduction de « Smart Mobs ». M2 Editions. Paris. ISBN 2 9520514 2 9
- [Riekkilä et al., 2005] Riekkilä J, Salminen T, Hosio S & Alakärppä I. Requesting services by touching objects in the environment. Proc. 11th International Conference on Human-Computer Interaction, Las Vegas, Nevada, USA, 2005.
- [Röcker et al., 2005] Röcker C., Janse M.D., Portolan N., Streitz N. User requirements for Intelligent Home Environments : A scenario-driven approach and empirical cross-cultural study. Joint sOc-EUSAI Conference (Smart Objects Conference - European Symposium on Ambient Intelligence), Grenoble, France. pp. 111-116
- [Römer et al., 2004] Kay Römer, Thomas Schoch, Friedemann Mattern, Thomas Dübendorfer. Smart Identification Frameworks for Ubiquitous Computing Applications. Wireless Networks, Vol. 10, No. 6, pp. 689-700, 2004.
- [Römer et Domnitcheva, 2002] Römer K., Domnitcheva S. Smart Playing Cards : A ubiquitous computing game. Persona land Ubiquitous Computing 6, 371-377, 2002
- [Rousseau et al., 2003] Rousseau F., Oprea J., Paun L.S., Duda A. Omnisphere: a Personal Communication Environment. Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS'03).
- [Ruffer et al., 1995] Ruffer S.M., Yen D., Lee S. Client/Server computing technology: A framework for feasibility analysis and implementation. International Journal of Information Management, Vol. 15, N° 2, pp. 135-150, 1995.
- [Saar et al., 2004] Saar S., Stuz M., Thomas V. Towards intelligent recycling: a proposal to link bar codes to recycling information. Resources, Conservation and Recycling 41. pp. 15-22. 2004
- [Sahin, 2004] Sahin E. A qualitative and quantitative analysis of the impact of Auto ID technology on performance of the supply chains. Thèse de doctorat de l'Ecole Central Paris. France. 2004
- [Sanz et Alonso, 2001] Sanz R., Alonso M. Corba for Control Systems. Annual Reviews in Control 25: 169-181, 2001.

-
- [Schieder et al., 2004] Schieder A., Eggert L., Papadoglou N., Pittmann F. Components and concepts of the Ambient Networks Architecture. Wireless World Forum Meeting 12 4th – 5th November 2004. Canada.
- [Schilit et al., 1994] Schilit B., Adams N., and Want R. Context-Aware Computing Applications. In Proceedings of the Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, December 1994. Pages 85-90. IEEE Computer Society.
- [Schlueter, 2003] Schlueter Langdon C. Information systems architecture styles and business interaction patterns: Toward theoretic correspondence. Information Systems and e-Business Management. Springer-Verlag 2003. 1:283–304.
- [Schmidt, 2000] Schmidt A. Implicit Human Computer Interaction Through Context. Personal Technologies Volume 4(2&3), June 2000. pp191-199.
- [Seen, 2000] Seen P. Challenges in microelectronics. Requirements for the multimedia era. C. R. Acad. Sci. Paris, t. 1, Série IV, p. 951–964, 2000. Solids, fluids: electronic and optical properties.
- [Segara et al., 2003] Segarra M.T., Keryell R., Plazaola A., Thépaut A., Mokhtari M. Ametsa : Un système de contrôle de l'environnement domestique générique fondé sur UPnP. SETIT (Sciences Electroniques, Technologies de l'Information et des Télécommunications), Sousse, Tunisie, 17-21 mars, 2003.
- [Siegemund et Flörkemeier, 2003] Siegemund F., Flörkemeier C. Interaction in Pervasive Computing Settings Using Bluetooth-enabled Active Tags and Passive RFID Technology together with Mobile Phones. Proceedings of PerCom 2003 (IEEE International Conference on Pervasive Computing and Communications), pp. 378-387, 2003.
- [Siegemund, 2004] Siegemund F. Cooperating Smart Everyday Objects – Exploiting Heterogeneity and Pervasiveness in Smart Environments. PhD dissertation at Swiss Federal Institute of Technology Zurich (ETH Zurich). 2004.
- [Singh et al., 2005] Singh S., Puradkar S., Lee Y. Ubiquitous computing: connecting Pervasive computing through Semantic web. ISeB. 2005.
- [Smirnov et al., 2004] Smirnov A., Pashkin M., Chilov N., Levashova T., Krizhanovsky A. Web-based intelligent support for logistics management. INCOM 2004: 11th IFAC Symposium on Information Control Problems in Manufacturing. Salvador da Bahia, Brazil. 2004.
- [Smit, 2004] Smit H. Technology Survey: Universal Plug and Play. ITEA (Information Technology for European Advancement) 02014 Project. 2004.
- [Sudarsan et al., 2005] Sudarsan R., Fenves S.J., Sriram R.D., Wang F. A product information modeling framework for product lifecycle management. Computer-Aided Design 37: 1399–1411, 2005.
- [Sun Microsystems, 1999] Sun Microsystems. Jini™ Architectural Overview. Technical White Paper. 1999. Palo Alto, California.
- [Sydänheimo et al., 2005] Sydänheimo L., Ukkonen L., Kivikoski M. Effects of size and shape of metallic objects on performance of passive radio frequency identification. International Journal of Advanced Manufacturing Technology. Accepted: 23 April 2005.
- [Tan, 2001] Tan K.C. A framework of supply chain management literature. European Journal of Purchasing & Supply Management 7:39-48, 2001.
- [Theiss et al., 2005] Theiss A., Yen D.C., Ku C-Y. Global Positioning Systems: an analysis of applications, current development and future implementations. Computer Standards & Interfaces 27: 89–100, 2005.

- [OMG UML, 2004] Objet Management Group - UML. Unified Modeling Language: Superstructure. version 2.0 formal / 05-07-04.
- [UPnP Forum, 2003] UPnP Forum. UPnP™ Device Architecture 1.0. Technical Report Version 1.0.1, Decembre 2003.
- [Urnes *et al.*, 2001] Urnes T., Hatlen A, Malm P., Myhre O. Building Distributed Context-Aware Applications. *Personal and Ubiquitous Computing* (2001) 5:38-41.
- [Vacherand, 2005] Vacherand F. New technologies for contactless microsystems. Joint sOc-EUSAI Conference (Smart Objects Conference - European Symposium on Ambient Intelligence) , Grenoble, France, 2005.
- [Valckenaers *et al.*, 1998] Valckenaers P., Van Brussel H., Wyns J., Bongaerts L., Peeters P., Designing of holonic manufacturing systems, *Robotics and Computer-Integrated Manufacturing*, 14, pp. 455-464, 1998.
- [Van Loenen, 2003] Van Loenen Evert J. On the role of Graspable objects in the ambient intelligence paradigme. *Proceedings of sOc'2003 (Smart Objects Conference)*, Grenoble, 2003.
- [Vanthournout *et al.*, 2005] Vanthournout K., Deconinck G., Belmans R. A taxonomy for resource discovery. *Personal and Ubiquitous Computing*, 9: 81–89, 2005.
- [VDI 4472, 2005] VDI 4472:2005. Requirements to be met by transponder systems for use in the supply chain – General. 2005
- [W3C OWL, 2004] OWL Web Ontology Language Guide. <http://www.w3.org/TR/owl-guide/>
- [W3C SOAP, 2003] SOAP Version 1.2 Part 0: Primer. W3C Recommendation 24 June 2003. <http://www.w3.org/TR/soap12-part0/>
- [W3C WS, 2004] Web Services Architecture. W3C Working Group Note 11 February 2004. <http://www.w3.org/TR/ws-arch/>
- [W3C XML, 2004] Extensible Markup Language (XML) 1.0 (Third Edition). W3C Recommendation 04 February 2004. <http://www.w3.org/TR/2004/REC-xml-20040204/>
- [W3C XML-E, 2001] XML Encryption WG. <http://www.w3.org/Encryption/2001/>
- [W3C XML-S, 2003] XML Signature WG. <http://www.w3.org/Signature/>
- [Walk, 2004] Walk Eldor. Standard RFID pour la logistique: situation actuelle. *DATAcollection* Avril 2004. pp. 28-32.
- [Wang *et al.*, 2004] Wang H., Zhexue Huang J., Qu Y. Xie J. Web services: problems and future directions. *Web Semantics: Science, Services and Agents on the World Wide Web* 1 (2004) 309–320.
- [Wang *et al.*, 2006] Wang N., Zhang N., Wang M. Wireless sensors in agriculture and food industry – recent development and future perspective. *Computers and Electronics in Agriculture* 50, 1-14, 2006.
- [Wartenberg *et Snyder*, 2003] Wartenberg N., Snyder S. Introduction to Bar Codes for the Automated Laboratory. *Journal of the Association for Laboratory Automation*, Volume 8, Issue 5, October-November 2003, Pages 51-58.
- [Weiser *et Brown*, 1997] Weiser M, Brown J.S. *The Coming Age of Calm Technology, Beyond Calculation: The Next Fifty Years of Computing*. P. Denning and R. Metcalfe, Editors, Springer-Verlag, Inc., New York (1997).
- [Weiser, 1991] Weiser M. The computer for the 21st century. *Scientific American* 265. N°3, 1991. 94-104.

-
- [Weiser, 1999] Weiser M., Gold R., Brown J.S. The origins of ubiquitous computing research at PARC in the late 1980s. *IBM Systems Journal*, Vol 38, N° 4, 1999.
- [Willard, 2006] Willard B. UML for systems engineering. *Computer Standards & Interfaces*. Article in Press 2006. Article accepted 30 December 2005.
- [Wong et al., 2002] Wong C.Y., McFarlane D., Zaharudin A.A., Agarwal V., The Intelligent Product Driven Supply Chain. *IEEE International Conference on Systems, Man and Cybernetics*. Hammamet Tunisia. 2002.
- [Workshop DC, 2004] The Disappearing Computer Workshop Report. European Commission – US National Science Foundation Strategic Workshop. Vienna, Austria, 23-24 April 2004.
- [Yoo et Lyytinen, 2005] Yoo Youngjin, Lyytinen Kalle. Editorial Social Impacts of ubiquitous computing: exploring critical interactions between mobility, context and technology. A special issue for *Information and Organization*. *Information and Organization* 15, 91-94, 2005.
- [Yoshimura et Hasegawa, 2003] Yoshimura T. et Hasegawa H. Comparing the precision and accuracy of GPS positioning in forested areas. *The Japanese Forestry Society and Springer-Verlag Tokyo* 2003. *J For Res* 8:147–152: 2003.
- [Zemirli et al., 2005] Zemirli N., Tamine L.L. & Boughanem M. Accès personnalisé à l'information : Proposition d'un profil utilisateur multidimensionnel. *7th International Symposium on Programming and Systems (ISPS) Algiers* May 2005.
- [Zhang et al., 2005] Zhang W. Y., Tor S. Y., Britton G. A. Managing modularity in product family design with functional modelling. *International Journal of Advanced Manufacturing Technology*. Article in Press. Accepted: 20 May 2005.
- [Zhu et al., 2002] Zhu F., Mutka M., Ni L. Classification of Service Discovery in Pervasive Computing Environments. Technical Report, Michigan State University, East Lansing, 2002.
- [Zhu et al., 2005] Zhu F., Mutka M., Ni L. Service discovery in pervasive computing environments. *IEEE Pervasive Computing*. October - December 2005 (Vol. 4, No. 4) pp. 81-90.

Résumé de thèse

« Contribution à la Modélisation et à la Gestion des Interactions Produit-Processus dans la Chaîne Logistique par l'Approche Produits Communicants »

Dans le domaine de la chaîne logistique, nous constatons des besoins croissants d'information et d'interactions entre produits, processus et clients, et ceci durant le cycle de vie du produit. Cela entraîne le besoin, au niveau du produit, de nouvelles capacités de communication, de gestion de l'information, de perception et d'action avec son environnement physique. Ces besoins ont engendré le concept de produit ou objet communicant. Un produit avec ces nouvelles capacités pourra interagir avec d'autres entités physiques ou informationnelles dans son environnement et apporter des transformations significatives sur la gestion de la chaîne logistique. Le travail présenté dans cette thèse vise à analyser et contribuer à appliquer la notion d'objet communicant aux produits physiques dans le domaine de la chaîne logistique. L'approche proposée considère un produit comme un demandeur ou un fournisseur de services. La méthodologie proposée de gestion des produits communicants s'appuie sur la caractérisation d'une architecture de services ambiants devant permettre de gérer les services d'un produit d'une façon automatique et ubiquiste. Nous avons choisi l'architecture UPnP pour gérer les services des objets communicants. La communication directe avec le produit est supportée par les méthodes d'identification automatique RFID. Nous nous sommes appuyés sur le formalisme standard de modélisation UML afin de modéliser les interactions et les services associés à un produit physique. Comme résultat, nous avons élaboré des démonstrateurs de laboratoire validant la faisabilité de notre proposition méthodologique de gestion de la chaîne logistique par les objets communicants.

MOT-CLES : Objets Communicants, Technologie d'identification automatique RFID (*Radio Frequency Identification*), Architecture de Services Ambiants UPnP (*Universal Plug and Play*), Modélisation, UML (*Unified Modeling Language*), Chaîne Logistique.

Thesis Summary

“A Contribution to Modelling and Management of Product / Process Interactions within the Supply Chain based on Communicating Object's Approach”

Nowadays, in the manufacturing field, it's necessary to manage, in real-time, the information related to the interactions between products, processes and customers along the product lifecycle. These requirements generate the communicating object concept (Smart Object), which represents a physical product equipped with memory, perception, communication, and action and decision capabilities in order to interact in its physical environment. Thus, a product with these new capabilities will be able to interact with other physical or informational entities and to bring significant transformations in the supply chain management. The work presented in this thesis aims to analyze and contribute to the implementation of the communicating object concept in the supply chain. The proposed approach considers a product as a service provider or a service requester. The proposed methodology is based on the ambient services architecture concept in order to manage product's services in an automatic and ubiquitous way. We have chosen the UPnP technology (Universal Plug and Play) to manage the services of the communicating objects. In our approach, the direct communication with the products is supported by means of the automatic identification of the products (RFID Technology). We have used UML (Unified Modelling Language) with the aim to model the interactions and the services associated with the physical products. Finally, we have created laboratory prototypes to validate the feasibility of our methodological proposition in order to manage the supply chain based on services interactions of communicating objects.

Keywords: Communicating Objects, RFID technology, UPnP Technology, Modelling Process, UML (*Unified Modeling Language*), Supply Chain.