REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mohamed Khider – BISKRA
Faculté des Sciences Exactes, des Sciences de la Nature et de la Vie
Département d'informatique

N° d'ordre : R.T.I.C. 18 /M2/2019

Mémoire

Présenté pour obtenir le diplôme de master académique en

Informatique

Parcours : **R.T.I.C.**

L'Internet social des objets " SIoT "

Par :

MLLE : BEDDA SAMAH

Soutenu le 07 juillet 2019, devant le jury composé de :

Monsieur HOUHOU Okba	MAA	Président
Monsieur HAMIDA Ammar	MAA	Rapporteur
Monsieur BELAICHE Hamza	MAA	Examinateur

Dédicaces

À vous mes parents ;
À vous mes frères et sœurs;
À vous mes chers (es) amis(es) ;
À vous mes chers collègues ;
À tous (toutes) ceux (celles) qui m'ont aidée ;
À tous (toutes) ceux (celles) qui me sont chers(es) ;
À tous (toutes) ceux (celles) que j'ai omis (es) de citer ;
Je dédie ce modeste travail !

Remerciements

Louange à dieu, seigneur des mondes qui m'a comblée de ses bienfaits et m'a aidée à réaliser ce modeste travail.

- Je remercie sincèrement Monsieur. HAMIDA Ammar qui m'a assistée et soutenue afin de réaliser cet ouvrage.
- J'exprime ici ma profonde gratitude au président et aux membres du jury d'avoir honoré et accepté de juger mon travail.

Enfin, que toutes ou tous ceux qui ont apporté ne serait-ce qu'une petite contribution à l'élaboration de ce mémoire puissent y trouver l'expression de mes sentiments de vive reconnaissance et mes souhaits les plus distingués sans oublier de souhaiter le succès à tous les étudiants de la promotion Master (2019).

Sommaire

Liste des figures

Liste des tableaux

Glossaire

Abr.	Sens	Pages
A.P.I	Application Programming Interface	43-51
A.R.P.A.Net	Advanced Research Projects Agency Network	18
C.I.S.C.O	Computer Information System Company	1-2
C-LOC	Relation de Co-localisation	9-10
C-LOR	Co-Location Object Relationship	10-13-46
CM	Control Manager	46-47-48
C-TRA	Relation de Co-travail	9-10
C-WOR	Co-work object relationship	10-13-46
D.H.T.	Distributed Hash Table:	5-34
D.O.S.	Denial of Service	25-38-50
D.S.	Demandeur de Service	13-26-27-28-29-32-33-35-37-38-39
FM	Friendship Manager	46
F.S.	Fournisseur de Service	13-16-26-27-29-32-33-35-37-38-39-45
I.A.	intelligence artificielle	51
ID	Identifiant objet	12-13-15-43-44-45-46-47-48-49
IDM	ID Manager	46-47-48
IM	Information Manager	46-47-48
I.O.T	Internet Of Things	1-2-3-4-5-6-7-15-24-27-28-29-31-34-40-41-42-43-45
I.P	Internet Protocol	3-16-19-20
I.P.S.O	Internet Protocol for Smart Objects	3
I.R.I.S :	Internet Routing in Space	1
M2M	Machine to Machine	4
M.F.T.P.	Multisource File Transfert Protocol	*24*
OC	The owner control	15-43-44-45
OOR	Ownership object relationship	10-13-46
OWL-DL	Ontology Web Language Description Logics	46-47
PAR	Relation de Parenté	9-10
P.K.I.	Public Key Infrastructure	24
POR	Parental object relationship	10-13-40-47
PRO	Relation de Propriété	9-10
PSM	Privacy and Security Manager	46-47-48
Q.O.S.	Quality of Service	5-6-19-26-27-31-33-34-37
RDF	Resource Description Framework	47
R.F.I.D	Radio Frequency Identification	1-4-10-16-44
RM	The relationship management	15-43-44-45-46-47-48
SC	The service composition	16-43-44-45
SD	The service discovery	16-43-44-45
S.I.O.T	Social Internet of Objects	4-7-8-9-12-14-15-16-24-25-28-29-31-32-33-39-40-41-42-44-45-46-47-48-49-50-51
SOR	Social Object Relationship	9-10-11-13-40
SPAROL	SPARQL Protocol and RDF Query Language	47
S.S.C.M.	Service Supply Chain Management	5
T.I.C	Technologies de l'information et de la communication	2
TM	The trustworthiness management	16-43-44-45-46-47-48
U.D.P.	User Datagram Protocol	23
WBAN	Wireless Body Area Network	44
WLAN	Wireless Local Area Network	44
W.S.N	Wireless Sensor Network	1-4

Généralités

1. Introduction

Les hommes ont rêvé. Ne devrait-il pas y avoir un réseau qui oblige tous nos appareils à collaborer à tout moment, à converser spontanément entre eux et avec le reste du monde et à constituer ensemble une sorte d'ordinateur virtuel unique.

Après le Web (les années 1990) et l'Internet mobile (les années 2000), nous nous dirigeons vers la troisième et potentiellement la plus perturbante phase de la révolution de l'Internet qui couvre maintenant des endroits jusqu'alors inaccessibles. Des patients ingèrent même des dispositifs connectés qui aident les médecins à diagnostiquer certaines pathologies et à en déterminer les causes **[1] ,** alors, chacun d'entre nous, des plantes, des animaux et des sites géologiques porteront des centaines d'implants d'ordinateur sur tout le corps [2] L'Internet est en train de conquérir l'espace grâce au programme IRIS de Cisco.

Au fil du temps, on a intégré l'ordinateur dans différents objets qui peuvent se connecter et communiquer entre eux. Elles sont conçues pour exécuter, puis penser, et aujourd'hui, elles apprennent à percevoir, sentir et réagir. C'est dans ce contexte que nous orientons notre recherche sur le concept de lier les objets à l'Internet, connu de nos jours comme "Internet of Things" (IoT) dont Nous entendons de plus en plus parler. **[3] [4] [5]**.

2. Problématique, Objectifs et Contributions

Pour répondre au besoin de l'utilisateur qui cherche à obtenir le bon service au bon moment et au bon endroit, on lance des défis à la mise en œuvre réussie de l'IoT. Le souci est d'arriver à combiner le monde physique et le monde virtuel, autrement dit, de travailler sur la standardisation de ce nouveau système en développant de nouvelles applications, de nouveaux protocoles de communication et d'environnements. Une littérature a proposé même un système permettant d'identifier et de localiser des objets en temps réel en utilisant **[6]**:

- **Le système RFID (***identification par radiofréquence***)** : une technologie d'identification automatique utilisée dans l'enregistrement et la réception de données;
- **Le protocole ZigBee** : une technologie de communication du réseau de capteurs sans fil (WSN :*Wireless Sensor Network*) à courte distance, à faible vitesse et à faible consommation d'énergie ;
- **Le Cloud Computing** : La zone géographique de couverture de leur système est très limitée et leur approche utilise toujours le principe client-serveur qui présente des failles, ce qui met en péril tout le système. D'ailleurs, Angulo Lopez et Jimenez Perez

ont proposé un cadre d'applications basé sur un système multi-agents pour simplifier le développement et l'intégration d'applications dans l'IoT, ce qui permet le partage de données et de services **[7].**

3. Hypothèse, Motivations et méthodologie

L'hypothèse principale qui sous-tend la présente recherche repose sur les limites de communication et de circulation des services dans le monde de l'IoT. Nous estimons alors qu'il est nécessaire de proposer une stratégie de collaboration pour partager les données.

Les avantages et les capacités de l'Internet ont donné naissance à l'IoT qui est considéré comme une véritable révolution dans le monde des T.I.C. Selon le géant de la réseautique Cisco, les appareils connectés atteindront le nombre de cinquante milliards en l'an 2020 **[8][9][10]** et intègreront notre vie personnelle et professionnelle dans ses différentes facettes d'où l'importance de mettre l'accent sur la nécessité du partage collaboratif dans l'IoT afin de créer un système à part entière qui répond à nos besoins.

L'existence d'un lien entre un objet et l'Internet est le fondement principal du fonctionnement de l'IoT. Ainsi, pour que cette relation soit à la hauteur de nos besoins, il faut adopter un nouveau système de communication basé sur la collaboration et le partage, surtout avec une multitude d'appareils connectés à partir de différents domaines.

Alors, comment gérer la multitude de données de l'IoT pour nous assurer de son bon fonctionnement? À cet effet, nous sommes basés sur les connaissances et les technologies déjà existantes pour situer la question principale de recherche.

4. Présentation technique du document

Le document est organisé en quatre chapitres et une conclusion générale :

- **Chapitre 1** : Comprend un aperçu général sur l'IoT ;
- **Chapitre 2** : Exposé du paradigme SIoT en commençant par sa définition, son mode de fonctionnement puis ses différentes structures, et nous l'appuyons enfin par la notion de confiance, ses propriétés, les attaques liées et les principales contraintes de sa gestion puis une classification de ses différents modèles.
- **Chapitre 3** : nous proposons une conception du paradigme SIoT, puis nous concluons le document en citant les différents défis à relever.
- **Chapitre 4** : nous proposons une implémentation SIoT. Nous avons présenté le logiciel de simulation, décrit son architecture, méthode de son installation, ses principaux fichiers et enfin méthode d'implémentation d'un réseau.

Chapitre1: L'internet des objets

1. Introduction

1.1. Définition de l'Internet of Things (IoT)

L'IoT est un réseau qui relie et combine des objets avec l'Internet en suivant des protocoles qui assurent leur communication et échange d'informations à travers une variété de dispositifs **[11].** Il se définit comme «un réseau de réseaux» qui permet d'identifier sans ambiguïté des entités numériques et des objets physiques et ainsi, de pouvoir récupérer, stocker, transférer et traiter les données sans discontinuité entre les mondes physiques et virtuels via des systèmes d'identification électroniques normalisés et unifiés et des dispositifs mobiles sans fil **[12].**

1.2. Un peu d'histoire

- **1832** : le **baron Schilling** émit une émission électromagnétique en Russie ;
- **1833**: **C. F. Gauss** et **W. Weber** transmettent à une distance de 1,2 Km à Göttingen;
- **1950** : **Alan Turing** avait déclaré qu'il est possible d'introduire des organes sensoriels dans les machines et les utiliser pour apprendre et enseigner l'anglais ;
- **1999** : **Kevin Ashton** invente le terme Internet des objets (IoT) **[15];**
- **2003** : Création du premier objet connecté, la lampe DIAL ;
- **2007** : Apparition des Smartphones ;
- **2008** : Création des adresses IPSO, IP qui permettent aux objets d'interagir entre eux.

1.3. Exemple illustratif

Supposons que nous voulons développer une application Web et mobile pour offrir des services météo collaboratifs. Ainsi, il est pertinent de bénéficier du concept de l'IoT pour récolter les données météorologiques dans différents endroits à des intervalles réguliers. Les informations recherchées exigent à la fois l'interconnexion et la communication entre l'ensemble des thermomètres installés, qui effectuent des échanges une fois connectés entre eux à travers l'Internet. Si l'un d'eux ne détient pas une information demandée sollicitera les autres avec lesquels il est connecté. Grâce à ce mécanisme, nous rêvons l'information voulue : c'est ainsi que cet exemple de service météo collaboratif peut devenir une réalité.

1.4. Domaines d'application de l'IoT

L'affirmation que "*L'IoT va littéralement transformer notre société*" vous semblerait utopiste, pourtant, il suffit de voir à quel point les choses ont évolué depuis l'arrivée de

l'iPhone. Récemment, l'idée que "la convergence de l'IoT et Réseaux sociaux est possible, voire souhaitable" gagne du terrain est due au fait que le paradigme de SIoT aurait de nombreuses implications souhaitables dans un monde futur peuplé d'objets intelligents imprégnant notre vie quotidienne. Ainsi, L'IoT consiste en un monde de données énormes, qui contribueront à répondre aux problèmes d'aujourd'hui dans beaucoup de domaines **[21].** Donc, il serait possible de trouver le concept de l'IoT n'importe où, n'importe quand et à la disposition de tout le monde. (Gubbi *et al*. ,2013) a classé son application en 4 domaines: le domaine personnel, l'environnement, l'infrastructure et les services publics et le domaine du transport **[20].** Actuellement, l'IoT est implémenté dans beaucoup de domaines tels que :

- Bâtiments intelligents ;
- Médecine ;
- Agriculture, alimentation ;
- Médias, divertissement ;
- Commerce, Gestion de la chaîne logistique ;
- Télécommunication, Transport, Aérospatiale ;
- Fabrication,, Recyclage, Pétrole et gaz ;
- Sécurité, assurance, environnement.

2. Fonctionnement de l'IoT

2.1. Technologies utilisées pour son fonctionnement

L'IoT nécessite plusieurs systèmes technologiques pour son fonctionnement. Afin de permettre l'interconnexion de différents objets intelligents via l'Internet, nous ne mettons l'accent que sur quelques-unes qui lui sont clés **[11**]. Ces technologies désignent diverses solutions techniques qui permettent d'identifier des objets, de capter, stocker, traiter, et transférer des données dans les environnements physiques, mais aussi entre des contextes physiques et des univers virtuels **[12]** sont:

- **RFID (** *Radio Frequency Identification) :* Englobe les technologies utilisant les ondes radio pour identifier des objets ou des personnes. Cette méthode permet grâce à une étiquette qui émet des ondes radio **[22]** de mémoriser et de récupérer des informations à distance ou pour les identifier à distance **[11]** et les transférer des étiquettes aux objets.
- **WSN** (*Wireless Sensor Network*) : est un ensemble de nœuds organisés en un réseau coopératif qui communiquent sans fil, où chacun possède une capacité de traitement et peut contenir différents types de mémoires : un émetteur-récepteur (Radiofréquence

RF) et une source d'alimentation, comme il peut aussi tenir compte des divers capteurs et des actionneurs. Il constitue alors un réseau de capteurs sans fil nécessaire au fonctionnement de l'IoT **[23].**

- **M2M (***Machine to Machine***) :**c'est l'association des technologies de l'information et de la communication avec des objets intelligents dans le but de donner à ces derniers les moyens d'interagir sans intervention humaine avec les systèmes d'information **[24].**

2.2. Modèle collaboratif de partage de données

La naissance du phénomène de l'IoT nous ouvre la voie pour développer des applications et des modèles de collaboration et de partage de données qui n'étaient pas possibles auparavant et qui fait l'objet de plusieurs recherches actuellement. À travers l'IoT, on recherche le potentiel de nouvelles technologies et leurs impacts dans l'amélioration continue du monde réel. Ces trois exemples dénichés parmi les divers travaux réalisés par des chercheurs lèvent le voile sur des solutions à notre problématique:

- **Guinard et Trifa [25]** : Se sont concentrés sur l'intégration des dispositifs du monde réel sur l'internet, ce qui leur permet d'être facilement combinés avec d'autres ressources virtuelles et physiques ;
- **Spiess et al. [26]** *:* Se sont dirigés vers l'intégration effective de l'IoT dans les services des entreprises ;
- **Han et Zhang [11]** *:* Ont travaillé sur la combinaison IoT et Cloud Computing pour faire face à l'augmentation massive des données.

Le développement d'un modèle capable de simplifier le partage et la collaboration dans l'IoT s'est fait en trois plans distincts et succincts qui sont ses principaux axes.

2.3. Découverte de services

L'importance de la phase de découverte de services dans le cadre de l'IoT est donc un des premiers postulats de la raison pratique :

- **Varguez-Moo *et al*. [59] :** ont proposé une approche méthodologique de la découverte des services, qui pourrait être cernée par l'usage d'un algorithme d'apprentissage automatique ;
- **Paganelli & Parlanti [60]:** ont proposé un éventail de réponses basé sur une méthodologie d'approche P2P (*un modèle d'échange où chaque entité du réseau est à la fois client et serveur) ;*

- **Altmann *et al*. [61] :** ont proposé une approche sur la puissance des tables de hachage distribuées (DHT). Elle consiste à dénicher tous les fournisseurs de services, attribuer des valeurs de hachage pour chacun pour enfin résoudre les collisions de hachage.

2.4. Sélection de services

D'autres chercheurs ont contribué au développement et à la compréhension de ce plan:

- **Makhlughian *et al*. [62]** ont présenté un modèle de sélection global qui permet de classifier les services des fournisseurs tout en respectant le niveau de la QoS (Quality of Service) et la qualité fonctionnelle. Cette approche se déroule en trois phases :
  - **Première phase** : Phase de mise à niveau afin de normaliser les valeurs du QOS ;
  - **Deuxième phase** : Phase de classement des différents services selon la qualité ;
  - **Troisième phase** : Phase de détermination de l'utilité de chaque service.
- **Wang *et al.* [63],** ont opté pour une nouvelle approche composée de deux phases se nommant SSCM (Service Supply Chain Management):
  - **Première phase** : Représente le calcul de l'incertitude et de transformer les valeurs QOS quantitatives en valeurs de qualité de service qualitatives ;
  - **Deuxième phase** : représente la sélection de service.
- **Mohana & Dahiya [64]** ont proposé des approches de sélection de services, catégorisables en deux aspects :
  - **Sémantique** : la recherche du service est basée sur la connaissance des attributs ;
  - **Non-sémantique** : la recherche du service est basée sur la syntaxe.

2.5. Échange d'informations

L'échange d'informations est l'étape la plus cruciale de toute notre démarche méthodologique où nous présentons ces diverses solutions adoptées tels que:

- **Jeong *et al*. [65]** ont indiqué qu'un échange homogène de données est préalable à la réussite de l'interopérabilité ;
- **Xu *et al*. [66]** ont développé une méthode d'échanges de données basée sur les protocoles XML et les architectures des services Internet ;
- **Kubler *et al*. [67]:** soulignent que l'IoT nécessite plusieurs interfaces de communications avancées et normalisées pour faciliter les interactions entre tous types d'objets.

Figure 1: Le processus de partage de données.

Chapitre2 : L'Internet social des objets (SIoT)

1. Généralités

1.1. Définition de SIoT

L'Internet social des objets est un nouveau paradigme dans lequel **l'IoT fusionne avec les réseaux sociaux**, permettant aux personnes et aux appareils connectés d'interagir dans un cadre de réseau social afin de prendre en charge une nouvelle navigation sociale **[73]**.

Il est défini comme un IoT où les objets sont capables d'établir des relations avec d'autres objets liés à l'homme de manière autonome conformément aux règles définies par le propriétaire **[69]** pour publier des informations, des services, puis pouvoir les retrouver ou découvrir de nouvelles ressources.

L'illustration ci-dessous montre un réseau social attenant à un individu ou à une organisation favorise l'acquisition de diverses informations qui seront, un service intelligent à travers le processus d'interférences sophistiquées **[68].** Il est composé de 3 éléments:

- L'Internet des objets ;
- Le service collaboratif ;
- Le réseau social.

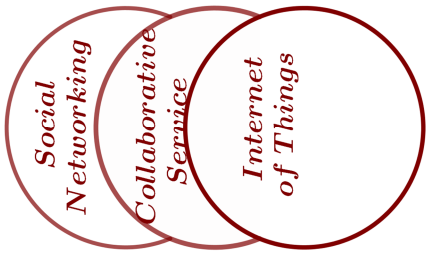

Figure 2: Les composants du SIoT

1.2. Concept du système SIoT

Le SIoT désigne un écosystème permettant aux personnes et aux objets intelligents d'interagir au sein d'une structure sociale basée sur des relations. Sa structure peut être configurée de façon à faciliter la navigabilité, effectuer la découverte d'objets et de services et garantir l'évolutivité comme dans les réseaux sociaux humains. Le niveau de confiance établi pour tirer parti du degré d'interaction entre les objets et les modèles de réseaux sociaux peuvent être adapté pour prendre en compte SIoT **[73]**.

Pour résumer, nous considérons que les éléments suivants font partie de l'architecture de SIoT **[14]:**

a. **Les acteurs (les utilisateurs et les Choses intelligentes)**: Chaque être humain et objet peut participer de manière égale en publiant des données et en recevant des commandes de contrôle permettant de gérer les données en cours de production qui peuvent être représentées sous forme de données de profilage ou simplement de réponses à des requêtes envoyées par des utilisateurs et / ou des périphériques ;
b. **Le système intelligent** : responsable de la gestion et de l'orchestration de l'ensemble des interactions entreprises par les acteurs telles que la gestion des services et des applications, les données et de contexte, la recommandations, la recherche et la découverte de services et gestion de contexte;
c. **L'interface** : Sert d'intermédiaire entre les acteurs et le système intelligent (les interactions avec le système ont lieu via l'interface). Elle permet la saisie de données et de requêtes et la sortie demandée telles que les Commandes de contrôle ou de services ;
d. **L'internet** : Il sert de moyen de communication et de lien entre les appareils intelligents avec leurs services et les utilisateurs.

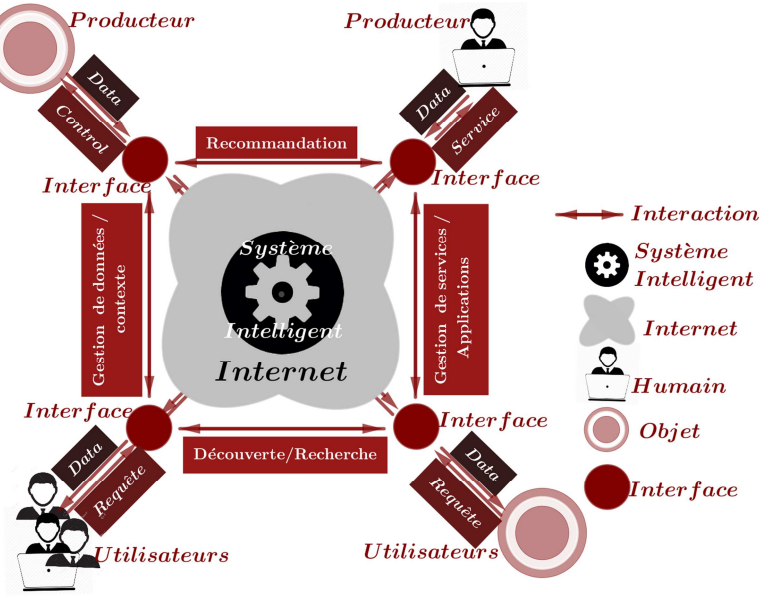

Figure 3**:** L'architecture du système SIoT

2. La structure sociale de SIoT.

2.1. Introduction

Le choix du meilleur ensemble de base de relations sociales peut être effectué en observant les modèles d'interactions entre les humains qui sont directement applicables aux comportements sociaux d'objets typiques mettant en œuvre des applications ubiquistes.

Nous sommes intéressés à établir et à exploiter les relations sociales entre les choses, afin de permettre aux objets d'explorer le SIoT et de découvrir des services et des ressources.

Les travaux de recherche menés par (Fiske et al., 1991), (George Mandler et al., 2015), etc…. dans les domaines de la sociologie et de l'anthropologie sociale sur les comportements sociaux entre les êtres humains, ont formalisé **les différentes formes de socialisation humaine et les interactions entre individus selon plusieurs types de relations**. Ces travaux ont proposé une classification des types de relations relatives aux objets.

Le SIoT est une combinaison des relations sociales inter-objets et entre leurs propriétaires. Ainsi, différents types de relations peuvent opérer entre eux, telles que: les relations sociales homme-homme, objet-objet et homme-objet. Elles sont créées et mises à jour sur la base des caractéristiques des objets et de leurs activités sans intervention humaine.

2.2. Hiérarchisation des types de Relations

Plusieurs travaux scientifiques (Procidano et *al*., 1983) (Zimet et *al*., 1988), ont analysé les comportements des individus au sein des différentes relations et leurs influences sur les interactions entre les individus qui sont souvent établies avec des niveaux de confiance différents selon le type de relation. Pareil aux êtres humains, les types de relations peuvent affecter les interactions sociales entre les objets surtout aux niveaux de la confiance entre les partenaires, et de la fiabilité des services **[50]**. Un objet ami avec lequel on a eu beaucoup d'interactions auparavant, ou un lien de parenté, est plus fiable qu'un objet inconnu, avec lequel il n'y a jamais eu d'interactions (Ashri et al., 2005). On est parvenu à la hiérarchisation des types de relations selon leurs niveaux de confiance suivant **[40]** :

a. **La relation de parenté (PAR)** : Plus forte en termes de fiabilité et de confiance, car la relation des membres d'une famille est caractérisée par un niveau de confiance élevé. Plus le lien est fort, plus les interactions sont solides et une confiance élevée.
b. **La relation de propriété (PRO) : est** un lien spécifique entre des objets du fait qu'ils appartiennent au même propriétaire. Toutefois, on considère que la relation PRO est caractérisée par un niveau de confiance inférieur à celui de la relation de parenté.
c. **La relation de Co-travail (C-TRA):** les objets n'ont généralement aucun lien de parenté, mais la coopération à un travail commun nécessite un certain niveau de confiance mutuelle. Alors, son niveau de confiance est inférieur à celui de PRO.
d. **La relation Sociale (SOR) :** elle est conditionnée par la rencontre opportuniste ou planifiée de leurs propriétaires (amis ou non). Étant donné l'incertitude des paramètres, nous considérons que son niveau de confiance est inférieur à celui de la CTRA.

e. **La relation de Co-localisation (C-LOC) :** elle s'établie entre des objets se trouvant dans le même environnement de communication ou réseau. C'est la relation la plus implicite entre les objets. Nous lui accordons le plus faible niveau de confiance.

À partir de la relation **C-LOC**, des objets peuvent décliner ensuite les autres relations, selon leurs paramètres et caractéristiques. En se basant sur une hiérarchie des types de relations présentée par ordre croissant dans le Tableau 1 **[30]:**

Types de relations	C-LOC	SOR	CTRA	PRO	PAR
Rang hiérarchique	❶	❷	❸	❹	❺

Tableau 1: Hiérarchie des relations entre objets

2.3. Les Types de relations entre objets

2.3.1. Approche sociologique de Mandler

Par analogie aux types de relations entre les individus, des travaux (Mandler et al., 2015) ont proposé une classification des types de relations relatives aux objets communicants sur la base de 5 types de relations qui sont **[40]:**

a. **Relation de Co-localisation (***Co-Location Object Relationship* **C-LOR)**: Établie lorsque des objets (tels que capteurs, actionneurs, étiquettes RFID, etc.) sont présents simultanément au même endroit (sur une machine, dans un atelier, une maison, une ville) pour proposer des services d'automatisation résidentielles ou industrielles;

b. **Relation de Co-travail (***Co-work object relationship* **C-WOR)**: Établie lorsque des objets coopèrent ensemble dans une même application ou un même processus, pour réaliser une tâche ou un but collectif (telles que : intervention d'urgence, télémédecine;

c. **Relation de Parenté** (*Parent Object Relationship* **: POR**): Établie lorsque des objets appartiennent à une même famille (objets similaires, même catégorie, construits à la même période, même fabricant, appartenant à un même lot). Elle est implémentée lors de la production de l'article, ne changera pas au fil du temps et ne sera mise à jour que par des événements d'interruption/d'obsolescence d'un périphérique donné ;

d. **Relation de Propriété (***Ownership object relationship* **OOR)** : Établie entre des objets hétérogènes appartenant au même utilisateur et interagissent entre eux (téléphones portables, lecteurs de musique, consoles de jeux, etc.). L'objet peut être porté par son utilisateur (personne, machine ou autre objet) qui peut stimuler l'interaction des objets.

e. **Relation Sociale (***Social Object Relationship* **SOR)** : Établie lorsque des objets entrent en contact au travers de la rencontre physique de leurs propriétaires de la même manière que les personnes qui échangent leurs contacts (n° de Tel, adresses e-mail, etc.). L'appareil, s'il y est autorisé, échange son profil social de manière autonome.

2.3.2. Approche sociologique de Fiske

L'anthropologue américain Alan Page Fiske (Fiske et al. 1991) a étudié la nature des relations humaines et leurs variations interculturelles, et a conçu une théorie des "modèles relationnels" dans laquelle il modélise les interactions sociales entre individus sur la base de quatre modèles relationnels, applicables à des objets qui sont **[41]:**

a. **Relation de partage communautaire *RPC (****Communal sharing relationships****):*** Les individus appartiennent à une communauté à laquelle ils contribuent avec ce qu'ils peuvent lui apporter et y prennent ce dont ils ont besoin ;

b. **La correspondance d'égalité *EC (****Equality matching****) :*** est basé sur des relations égalitaires caractérisées par une réciprocité en nature et un échange équilibré en volume où chaque individu est préoccupé de son équilibre;

c. **Relation de classement d'autorité *RCA (****Authority ranking relationships****)** : Les individus sont ordonnés selon certaines dimensions sociales hiérarchisées. Les relations entre individus sont ainsi assujetties à leurs rangs d'autorité asymétriques;

d. **Relation de prix du marché *PM* (***Market pricing relationships***)**: Elle est basée sur la proportionnalité de valeurs. Les interactions étant organisées en se référant à une échelle commune de valeurs des éléments échangés lors de l'interaction.

Relation	Socialité selon la théorie de Fiske	Socialité transposée aux objets communicants
Partage Communautaire (PC)	L'appartenance de chaque **individu** à une collectivité de partage l'emporte sur toute forme de caractère distinctif individuel.	Le partage des données, services et ressources entre tous les **objets** d'une communauté est admise de fait par tous, et sans condition.
Équilibre de Concordance (EC)	Les relations entre **individus** sont caractérisées par une réciprocité et des échanges équilibrés en valeur ou en quantité ou en nature.	Les relations entre **objets** sont quantifiées et contraintes par un équilibre en quantité ou en valeur des données, des services et des ressources.
Rang d'Autorité (RA)	Les individus sont ordonnés selon une hiérarchie, un statut, qui se traduisent par un classement d'autorité, et des relations asymétriques acceptées.	Les objets sont ordonnés selon un classement hiérarchique de leur rang autorité, imposant une subordination respectée dans leurs relations (maître, esclave).
Prix du Marché (PM)	Les interactions entre individus sont structurées par des coûts économiques, non forcément monétaires, et menées selon un mode d'échange marchand reposant sur une échelle de valeurs.	Les objets délivrent leurs services contre un coût prédéfini associé à chaque service selon un système de cotation partagé (jeton, bitcoin). Le principe de troc de services sans monétisation est permis.

Tableau 2: les quatres modèles relationnels élémentaires selon la théorie de Friske

2.3.3. Approche sociologique de Chen R. &Al.

Une autre structure a été proposée par Chen R. &Al **[72],** qui proposent un réseau SIoT basé sur 3 types de relations sociales reliant les propriétaires d'objets:

1. **Relation d'amitié** : Elle représente l'intimité ;
2. **Relation de contact social** : Elle représente la proximité;
3. **Relation de communauté d'intérêts** : fait référence à des connaissances ou expériences communes.

2.3.4. Approche sociologique de Ji Eun Kim & Al.

Les nouveaux types de relations [**71]** permettant aux humains et aux périphériques SIoT de collaborer, en complément des amitiés existantes sur les réseaux sociaux. 3 nouvelles relations entre appareils ont été rajoutées (Les 3 dernières du tableau ci-dessous) :

- **La parenté :** Pour le même modèle d'appareils du même fabricant ;
- **L'amitié :** Pour la relation entre des appareils appartenant à des amis (choses d'amis);
- **La propriété partagée :** les appareils appartenant au même utilisateur.

	Types de relations	**Définition de la relation**
1	Relation amicale (Friendship)	Relation entre utilisateurscomme dans les réseaux sociaux
2	Relation de propriété (Ownership)	Un appareil enregistré par son propriétaire
3	Relation de co-location (Co-location)	Utilisateurs et / ou appareils au même endroit
4	Relation de parenté (Kinship)	Appareils avec le même modèle et le même fabricant
5	Relation d'amitié (Thriendship)	Amitié entre choses / dispositifs d'amis
6	Relation de propriété partagée (SharedOwnership)	Appareils appartenant au même utilisateur

Tableau 3: Les différentes relations sociales (Ji Eun Kim et Al.)

2.4. Établissement des relations entre objets

L'établissement d'une relation dépend des caractéristiques et des paramètres associés à chaque objet, auxquels s'ajoutent des éléments supports de son aspect social, qui sont **[40]**:

- **Les attributs de l'objet :** Ils déterminent les caractéristiques propres de l'objet. Ils entrent en jeu lors d'une interaction sociale avec un autre objet:
  - **Identifiant objet (ID) :** Unique et défini par le fabricant de l'objet (UUID, MAC, Numéro de Série, ePC, …),
  - **Identifiant propriétaire (OID):** Les objets du même propriétaire ont le même OID.
  - **Code Famille (FC):** Code représentatif d'un groupement d'objets, possédant des caractéristiques fonctionnelles ou techniques communes. Une famille d'objets partage la même finalité ou le même usage.

— **Identifiant Application (AID)**: Code caractéristique de l'application, de l'activité ou du processus auquel l'objet participe (étape de fabrication, transport, surveillance environnement, …). Il permet d'identifier les objets qui travaillent ensemble pour la même application ou activité.

- **Services** : Tout objet est DS et/ou FS. Plusieurs paramètres influent sur l'exécution d'un service tels que:

– **Coût d'un service** : Un objet associe un coût pour autrui à chaque service. Ce coût est défini selon le domaine géographique, l'environnement de fonctionnement, la famille d'objets ou une application. Chaque objet possède un capital en monnaie électronique (ex. *bitcoin*) afin de régler les coûts de services qu'il demande.

– **Principes d'interaction sociale :** Les modèles relationnels sociaux régissent les principes d'interaction entre objets selon des protocoles d'interaction paramétrés. Ils sont exécutés et agréés par tous les objets en interaction, permettant des demandes et des échanges de services selon une approche socio- mimétique entre Objets.

- **Communauté sociale**: Tout objet crée un groupe social (Chen &al. 2016) qui est:

– Composé des objets avec lesquels il a eu des interactions et enrichi des types de relations et des niveaux de confiance correspondants ;

– Est mis à jour à partir d'un historique horodaté des interactions des objets et mémorise chacune d'elles (ID, services réalisés, …) et auquel se référera l'objet pour entamer une relation avec un autre objet (objet connu, niveau de confiance).

Un objet choisit une relation sur la base des caractéristiques de l'objet partenaire :

– Une fois la relation établie, les objets peuvent entrer en interaction opérationnelle ;

– Une fois que 2 objets se sont identifiés, ils sont par défaut en relation CLOC. Dès lors, d'autres relations peuvent s'établir selon les paramètres des objets ;

Note : Certains services peuvent être exclusifs à un type de relation. (Ex : par une relation PAR, un objet peut demander une configuration auprès d'un autre de la même famille).

```
IF (FC1==FC2)
  Relation=PAR
    Else IF (OID1==OID2)
      Relation=PRO
        Else IF (AID1==AID2)
          Relation=CTRA
            Else IF (OD1! =0) AND (OID2! =0)
              Relation=SOR
                Else Relation=CLOC
```

Schéma 1: Logique de détermination du type de relation

3. Modèle architectural de référence de SIoT:

3.1. Architecture suggérée par Nitti & Atzori

Nitti & Atzori **[70]** ont fourni un bloc de construction de base pour l'ensemble du domaine SIoT. Son architecture est composée de (figure 4**)**:

a. **Côté server à droite** : La partie server est divisée en trois couches principales :

- **La couche de base** : comprend des services pour les bases de données, les moteurs sémantiques et la communication ;
- **La couche des composants** : est utilisée comme outil hôte pour la mise en œuvre de composants satellites ;
- **La couche d'applications** : sert d'interface entre les objets et les humains. Elle est utilisée pour fournir des services de connexion.

b. **Côté client à gauche** : La partie client est divisée aussi en trois couches:

- **La couche d'objets** : est composée d'objets physiques ;
- **La couche d'abstractions d'objets** : agit comme une interface entre les périphériques connectés. Cette interface est contrôlée par des langages communs ;
- **3^{e} couche** : est composée d'un certain nombre d'agents et sa fonction consiste à établir des connexions entre les objets attachés et les serveurs à base de SIoT. La gestion des services fournit deux services: L'un est une interface, le second surveille et contrôle le comportement des objets.

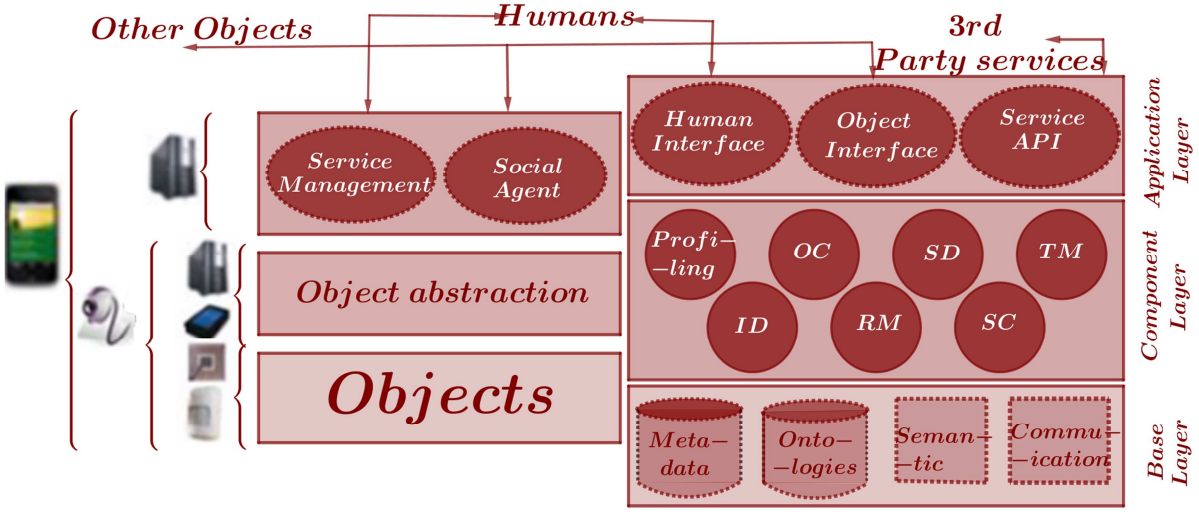

Figure 4: Architecture du système suggérée par Nitti & Atzori.

3.2. Architecture suggérée par Luigi Atzori & al.

Luigi Atzori, & Al**.[41]** , ont proposé, un modèle architectural à 3 couches qui sont:

- **Couche de détection** : consacrée à l'acquisition de données et à la collaboration des nœuds dans les réseaux locaux et à courte portée;
- **Couche réseau** : vise à transférer des données sur différents réseaux;
- **Couche d'applications**: où les applications IoT sont déployées avec les fonctionnalités middleware.

La figure5 montre l'architecture à trois couches. Les trois éléments de base du système proposé sont: **le serveur SIoT**, **la passerelle** et **l'objet [41]** :

a. **Côté serveur à gauche:** La partie server est composée de trois couches:

- **La sous-couche de base (***Base Layer***):** comprend la base de données pour le stockage et la gestion de données avec les descripteurs pertinents, la base de données d'ontologies, les moteurs sémantiques et les communications.
- **la sous-couche Composants** (*the Component Sub-layer*) comprend les outils qui implémentent la fonctionnalité principale du système SIoT tels que :
  - **La gestion des identifiants** *(The ID management) :* vise à attribuer un identifiant qui identifie universellement toutes les catégories possibles d'objets (UUID, MAC, N° de Série, ePC,…), et conserver les schémas d'identification d'objets existants habituellement définis par le fabricant de l'objet. Un protocole simple basé sur XML peut être mis en œuvre, permettant de spécifier le mécanisme d'identifiant adopté autre que l'identifiant lui-même. Ce système comprend: les adresses IPv6, le code universel de produit (UPC), le code électronique de produit (EPC), le code ubiquitaire (Ucode), OpenID, l'URI.
  - **Le profil d'objet** *(Object profiling* **OP***) :* vise à configurer manuellement et automatiquement une information (statique ou dynamique) sur les objets qui doivent être organisés en classes sur la base de leurs principales caractéristiques.
  - **Le contrôle du propriétaire** *(The owner control* ***OC****) :* Ce module permet la définition des éléments suivants en utilisant différents langages de politique de sécurité et de contrôle d'accès déjà disponibles:
    - ✓ Des activités pouvant être exécutées par l'objet ;
    - ✓ Des informations pouvant être partagées et auxquelles peut accéder l'objet;
    - ✓ Du type de relation à mettre en place.
  - **La gestion des relations** *(The relationship management RM)* : est le module clé du réseau. Sa tâche principale est de permettre aux objets de démarrer, de mettre

à jour et de mettre fin à leurs relations avec d'autres objets sur la base des paramètres de contrôle du propriétaire.

- **Découverte de service** *(The service discovery SD) :* Cet élément détermine les FSs requis de la même manière que les êtres humains en recherchant des amitiés et toute information contenue dans les services de réseau social.
- **Composition de service** *(The service composition SC)* : Active les interactions entre les objets ainsi que le service souhaité et trouvé par l'élément de découverte de services en s'appuyant sur les relations d'objets.
- **Gestion de confiance** *(the trustworthiness management TM)* : vise à comprendre comment doivent être traitées les informations fournies par les autres membres et savoir à quels services et objets faire confiance. La confiance est basée sur le comportement de l'objet et strictement liée au module de gestion des relations.

- **Sous-couche Interface** *(Interface Sub-layer):* les interfaces avec les objets, les humains et les services s'y trouvent. Elle peut être mappée sur un seul site, déployée de manière fédérée par différents sites ou déployée dans un nuage.

b. Côté Passerelle et objets à droite:

Quant aux systèmes de passerelles et d'objets, la combinaison des couches peut varier selon les caractéristiques du périphérique. Nous prévoyons l'un des 3 scénarios suivants **[41]:**

- **Premier scénario : un objet simple** (ex : l'étiquette RFID), un dispositif de détection équipé d'une fonctionnalité de la couche la plus basse est autorisé à envoyer des signaux à la passerelle qui est équipée des fonctionnalités des 3 couches.
- **Deuxième scénario : un dispositif** (ex: la caméra vidéo) est capable de détecter les informations physiques et d'envoyer les données associées sur une IP réseau. L'objet serait alors défini avec les fonctionnalités de la couche réseau autres que celles de l'application. Par conséquent, Une couche d'application sur un serveur avec la fonctionnalité de couche d'application de passerelle serait suffisante.
- **Troisième scénario** : **un objet intelligent** (ex: Smartphone) peut implémenter la fonctionnalité des 3 couches de sorte que la passerelle ne soit pas nécessaire. Mais certaines installations de communication sont nécessaires pour maintenir la connectivité de l'objet. Il dispose suffisamment de puissance de calcul pour effectuer toutes les opérations pour les 3 couches et nécessite une passerelle pour une connectivité réseau omniprésente.

Quel que soit le scénario mis en œuvre, la couche application englobe **les applications SIoT**, **l'agent social** et **l'agent de gestion de service** présentés ci-dessous **[41]** :

- **L'agent social** *(The social agent) :* Prévu pour la communication avec les serveurs afin de mettre à jour son profil et ses amitiés, de découvrir et de demander des services et d'implémenter les méthodes permettant de communiquer directement avec d'autres objets lorsqu'ils sont proches géographiquement ou lorsque la composition du service nécessite des communications directes entre les objets.
- **L'agent de gestion de services** *(the service management agent)* est responsable des interfaces avec les humains qui peuvent contrôler le comportement de l'objet lors de la communication au sein de son réseau social.

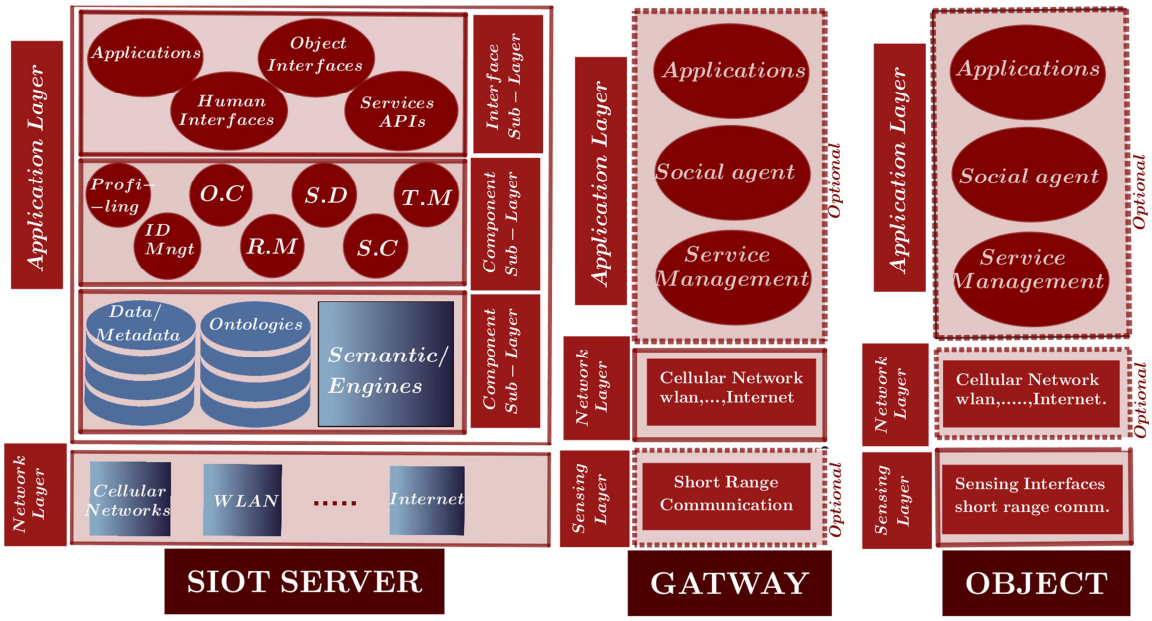

Figure 5: L'architecture proposée par Luigi Atzori, & Al

3.3. Le Peer-to-Peer ou Le Pair à Pair ou Le P2P

3.3.1. Introduction

Le monde de l'informatique est en effervescence autour d'un phénomène portant le nom barbare de **P2P**. Mal identifiée et mal considérée à ses débuts, mais, l'idée a beaucoup mûri dernièrement. Aujourd'hui, on parle sérieusement du P2P comme un modèle de communication capable de changer certaines approches de l'informatique en réseau.

3.3.2. Le problème soulevé

Imaginez: Un éditeur veut distribuer une version de démonstration d'un jeu très attendu. Habituellement, il va placer le fichier sur son site web afin que les internautes viennent le télécharger. Ce serveur web va donc être la seule source du fichier.

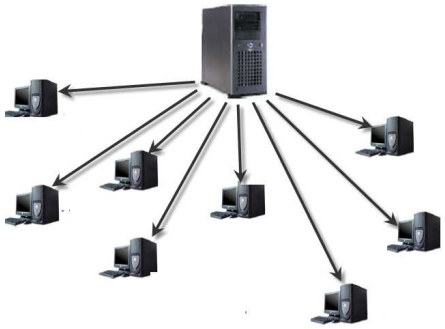

Figure 6: Première idée d'un P2P

Problème: il y aura énormément d'internautes qui téléchargent en même temps un fichier populaire. Comme ils veulent tous obtenir ce fichier, le serveur **voudra** en faire profiter tout le monde, mais, il a de plus en plus de mal à répondre aux demandes, au point de ne plus pouvoir répondre du tout ! Alors, personne n'y arrive à le télécharger car le serveur est surchargé de demandes. Cherchons donc un moyen de faire mieux !

3.3.3. La solution au problème soulevé : L'apparition du P2P

Le client/serveur s'est montré capable de répondre parfaitement aux attentes des clients qui est leur intégration au réseau. Mais son éloignement de la philosophie égalitaire avait donné naissance à l'Arpanet. Il faut vite y remédier ?

a. Le P2P : Un concept redécouvert par Napster

En 1998, l'étudiant Shawn Fanning (19 ans), écrit un logiciel qui permet d'échanger des fichiers audio de format mp3 via internet. C'est un assemblage de 4 composants: un navigateur web, un serveur de fichiers, un module de messagerie instantanée et un lecteur de fichiers mp3. Le serveur permet de mettre à jour en permanence la liste des fichiers audio partagés qui ne sont pas transférés au travers d'un serveur mais directement d'un utilisateur vers un autre utilisateur, définissant ainsi le principe d'échange du P2P.

b. La fin de Napster et l'arrivée du pur P2P

Devant les menaces de la justice pour violation des droits d'auteurs par Napster, puis l'interruption de son service sur décision judiciaire, de nouveaux logiciels similaires apparaissent et transfèrent l'ancienne fonction de serveur centralisé en milliers d'ordinateurs dans le monde: le « pur P2P » est né et revêt des nombreuses applications commerciales.

3.3.4. Présentation d'un P2P

a. Définition d'un P2P

L'informatique Peer-to-peer (Pair à Pair ou P2P) se définit comme le partage des ressources et des services informatiques par échanges direct entre systèmes par introduction

d'une relation d'égal à égal entre 2 ordinateurs. Ces échanges portent sur les informations, les cycles de traitement, la mémoire cache ou encore le stockage de fichiers sur disque.

Plusieurs facteurs participent à l'explosion du phénomène : la puissance, la largeur de bande passante et la capacité de stockage. Contrairement au modèle client/serveur, le P2P fait de chaque nœud du réseau une entité complète qui remplit à la fois le rôle de client et de serveur. Ex : Le jeu en réseau, l'email et même le téléphone sont pris pour des systèmes P2P.

b. Caractéristiques du P2P

Un vrai système P2P se reconnaît par les réponses positives aux 2 caractéristiques suivantes contrairement au modèle client/serveur:

1. Le système donne-t-il à chaque peer une autonomie significative ?
2. Permet-il à chacun de se connecter de manière intermittente avec des IP variables ?

Les caractéristiques importantes d'un système P2P sont directement liées à:

- La quantité et la qualité de données qui y sont disponibles ;
- Les ressources sont ajoutées à mesure que le nombre de peers du réseau augmente ;
- Localisation des ressources efficacement quelle que soit la taille du réseau.

c. Avantages et inconvénients d'un Peer to Peer

Le système P2P a ses avantages et ses inconvénients comme tout autre système:

- **Ses avantages :**
  - ✓ Les communications sont directes ;
  - ✓ Décentralisation ;
  - ✓ Passage à l'échelle ;
  - ✓ Connectivité intermittente ;
  - ✓ La réplication et redondance des données ;
  - ✓ Un nœud peut accéder directement à un ou plusieurs nœuds ;
  - ✓ Une machine tombée en panne ne remet pas en cause l'ensemble du système ;
  - ✓ Le réseau est faiblement couplé ;
  - ✓ Possibilité de créer des groupes.
- **Inconvénients**
  - ✓ Pas de QoS ;
  - ✓ Problèmes de sécurité ;
  - ✓ Les temps de localisation sont plus longs ;

3.3.5. Les différentes architectures Peer to Peer

L'application P2P comprend 2 modèles d'architecture **[43]** :

a. L'architecture centralisée (Le P2P assisté**) :** Tels que : Napster, Audiogalaxy:

- Mise en relation à partir d'un unique serveur qui recense les fichiers proposés par les différents clients puis la mise en liaison direct entre clients.
- Dispose de 2 types d'informations : celles sur le fichier ((Nom, taille, etc..) et celles sur le client (Nom, IP, nombre de fichiers, type de connexion, etc….).

❖ **Principe de fonctionnement :**

Un peer A se connecte au réseau en annonçant à un serveur quelles ressources mettra–t-il à sa disposition. Le serveur maintient un index des ressources disponibles sur le réseau à chaque instant. Si le peer A a besoin d'une ressource, il interroge le serveur qui localise la ressource sur un autre **peer B** connecté au réseau. Le serveur n'intervient plus à ce moment-là et le transfert de la ressource s'effectue directement entre le **peer A** et le **peer B**.

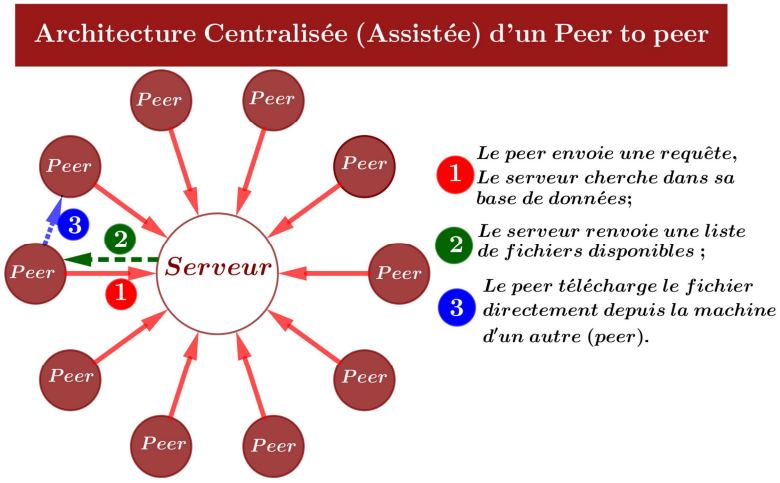

Figure 7: Architecture centralisée (assisté) d'un P2P

- **Ses avantages**
  - ✓ L'utilisation d'un index central qui permet de localiser les ressources et résoudre les requêtes rapidement et efficacement.
- **Ses inconvénients**
  - ✓ Problèmes de sécurité; de robustesse et de limitation de la bande passante;
  - ✓ Bloquer cette architecture pour déconnecter tous les utilisateurs et stopper le fonctionnement du réseau, mais les architectures hybrides permettent de contourner cette faiblesse ;

b. L'architecture décentralisée: Tels que Gnutella, CAN, Chord, Freenet, GNUnet, Tapestry, Pastry et Symphony:

- Chaque nœud est à la fois client et serveur ;
- Égalité entre tous les clients ;
- Liaisons établies de proche en proche ;
- Requêtes transférées et relayées ;
- Fichiers transférés directement au donneur ;
- Réseau en perpétuelle mutation.

❖ Mode de fonctionnement :

Un peer A équipé d'un programme spécifique, se connecte à un **peer B** équipé aussi de ce programme. **Le peer A** lui annonce ainsi sa présence sur le réseau. **Le peer B** relaie cette information à **tous les peers** auxquels il est connecté. Ceux-ci signifient alors leur présence au **peer A** et relaient l'information à leur tour aux peers auxquels ils sont connectés, et ainsi de suite jusqu'à " l'horizon visible3 " du **peer A**.

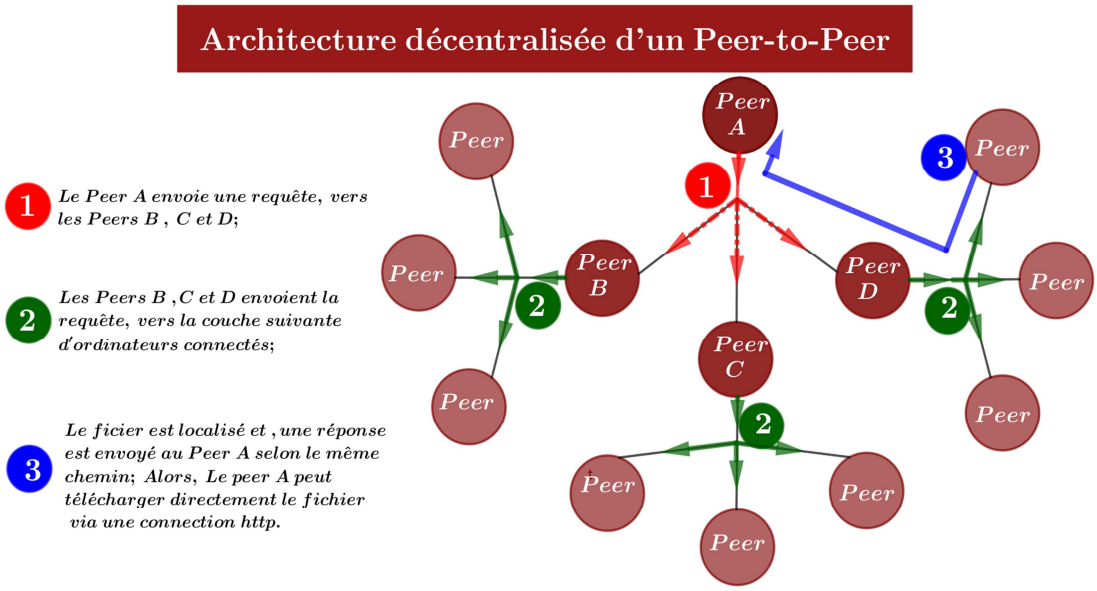

Figure 8: Architecture décentralisée d'un Peer-to-Peer

- **Ses avantages**
  - ✓ Plus robuste puisqu'il ne dépend pas d'un serveur;
  - ✓ Intermittence des connexions des nœuds;
  - ✓ La bande passante utile pour chaque requête croît exponentiellement quand le nombre de peers croît linéairement en raison de la façon dont sont transmises les requêtes (broadcast).
- **Ses inconvénients**
  - ✓ Lenteur des échanges de données due à des séries de broadcast ;
  - ✓ Il est facilement victime d'activités malicieuses;

c. Améliorations des architectures : Architectures hybrides

Le modèle assisté apporte certainement la meilleure des solutions en termes de rapidité de résolution des requêtes. Tandis que, le modèle décentralisé, lui, apporte la robustesse mais doit consommer beaucoup de ressources pour acheminer les requêtes. Les modèles hybrides font leur apparition afin d'apporter des solutions aux problèmes que connaissent les deux modèles suscités et de tirer parti de leurs points forts. Ils peuvent nécessiter plusieurs serveurs organisés de façon hiérarchique. La gamme des topologies hybrides imaginable est assez vaste et permet de répondre à de nombreuses problématiques.

❖ Architecture Hiérarchique

L'organisation de grappes de serveurs en anneau est largement utilisée dans le web. L'anneau permet de réaliser la répartition de charge et dilue le risque de défaillance localisée.

- **Avantage :** En connectant le réseau de peers à ces anneaux, on bénéficie de la simplicité d'un système assisté avec la robustesse d'un système décentralisé.

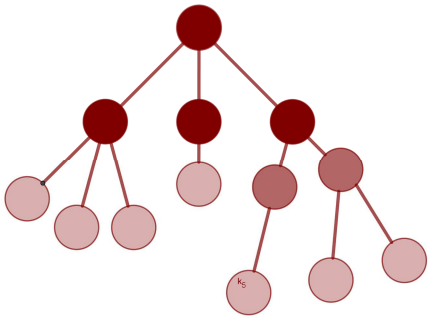

Figure 9: Architecture Hiérarchique

❖ Architecture en Anneau

L'organisation en anneau utilise une architecture combinant des systèmes centralisés au sein de systèmes décentralisés. Il remplace les serveurs dédiés centraux qui réalisent l'indexation du contenu par un grand nombre de super-nœuds afin de résoudre les problèmes de robustesse et améliorer la qualité de connexion avec le serveur.

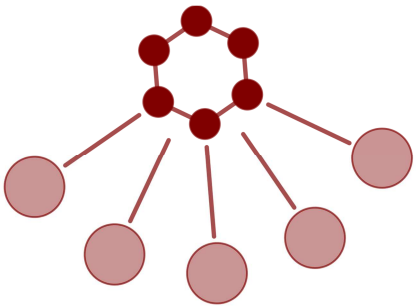

Figure 10: Architecture en Anneau

- **Ses avantages**
  - ✓ Elle dispose de bande passante et de puissance de calcul au-dessus de la moyenne. Elle peut être utilisée pour relayer des requêtes lorsque les ressources trouvées sont insuffisantes. Ce principe est populaire et améliore grandement les performances des réseaux décentralisés ;
  - ✓ Elle permet d'éviter la chute du réseau si une panne se produit sur un serveur, car il y a toujours un point de connexion valide aux serveurs ;
  - ✓ L'utilisation de plusieurs serveurs permet de mieux répartir les demandes de connexions et donc de limiter la chute de la bande passante ;
  - ✓ Chaque serveur peut avoir accès aux informations des clients connectés sur les autres serveurs. L'accès aux données partagées est donc totalement transparent.

❖ **Architecture avec super-nœuds**

Les super-nœuds (grands cercles) sont utilisés pour relayer des requêtes lorsque les ressources trouvées sont insuffisantes. Ils ont des fonctions de localisation et de publication des ressources. L'évolution des architectures logicielles P2P tendent vers cette vision. Ils ont des capacités identiques mais peuvent avoir de comportements différents et configurés différemment. Exemple : Kazaa.

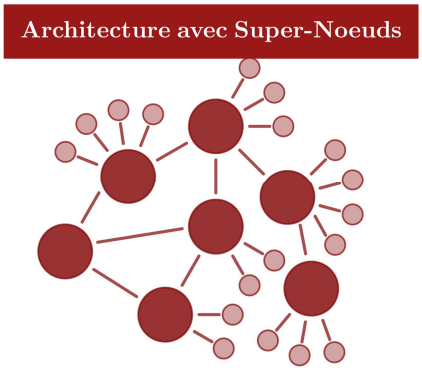

Figure 11: Architecture avec super-nœuds

4. La sécurité et la fiabilité sous SIoT

4.1. La sécurité et ses objectifs sous SIoT

La sécurité informatique est l'ensemble des moyens techniques utilisés qui visent à empêcher l'utilisation non-autorisée ou réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles **[17].**

Les solutions PKI (Public Key Infrastructure) ont permis d'échanger des informations en toute sécurité. Elles sont déjà utilisées pour traiter des problèmes similaires à ceux de

l'IoT, et pour sécuriser les appareils mobiles tels que les téléphones, les tablettes, les imprimantes et les points d'accès Wifi. Elle utilise des mécanismes de signature et certifie des clés publiques qui permettent de chiffrer, de signer des messages et des flux de données.

La sécurité Informatique d'une manière générale et les problèmes de sécurité et de vie privée dans SIoT d'une manière spéciale vise à assurer plusieurs objectifs tels que : **[17].**

1	**Authentification**	Est le processus de prouver une identité revendiquée par le moyen d'un seul ou plusieurs facteurs.
2	**Confidentialité**	S'assurer que l'information n'est accessible qu'à l'émetteur et le récepteur par chiffrement des données
3	**Intégrité**	Garantir la protection des données contre les modifications et les altérations non autorisées.
4	**Disponibilité**	Elle assure les entités autorisées d'accéder aux ressources réseaux afin d'éviter les attaques de type DOS.
5	**Non-répudiation**	Elle garantit qu'un message a bien été envoyé par un émetteur et reçu par un destinataire et aucun des 2 ne pourra nier son envoi ou sa réception.
6	**La réputation**	le degré de confiance attribuée à un nœud par l'ensemble des nœuds du réseau ou d'une partie **[18].**
7	**Outils de cryptographie**	Il est important de chiffrer les messages transmis aux nœuds afin d'assurer les confidentialités par méthode de chiffrement symétrique (même clé pour chiffrer et déchiffrer) ou par méthode asymétrique (avec différentes clés) [16]. **Fig12** illustre les fonctions de la gestion des clés **[13]**.

Tableau 4: Quelques objectifs qu'assure la sécurité informatique

Figure 12: Illustration des différentes fonctions de la gestion des clés

4.2. La confiance sous SIoT

4.2.1. Définition de la confiance :

Il existe différentes définitions de la confiance:

- Neisse et al. **[31] :** C'est la croyance mesurée par un trustor (qui accorde sa confiance) en ce qui concerne la compétence, l'honnêteté, la sécurité et l'habilité d'un trustée (à qui on fait confiance) dans un contexte donné.
- l'ITU-T X. 509 **[32]** : On dit qu'une entité fait confiance à une deuxième entité si et seulement si cette dernière se comporte exactement comme le prévoit la première.

4.2.2. Les éléments de la confiance

1	**Honnêteté**	le degré de fiabilité des recommandations fournies par une entité **[19].**
2	**Intérêt à la communauté**	Déterminé par l'appartenance à la même communauté;
3	**Interaction**	L'action menée en commun par des acteurs au sein d'un système **[27].**
4	**Contrôle d'accès**	Méthode de gestion de l'accès aux ressources**. [27].**
5	**Certificat**	Document électronique qui associe une clef publique à une identité en utilisant la notion de signature numérique **[27].**
6	**Réputation**	Comportement attendu d'une entité d'après des informations relatives à son comportement passé **[28].**
7	**Recommandation**	Une entité peut évaluer la qualité et la fiabilité d'une collaboration avec une autre entité et fournir cette information à une 3^{e} entité **[28].**
8	**La coopérativité**	Déterminé par l'interaction d'un nœud avec ses amis ou leurs amis;
9	**Logique floue**	Une extension de la logique booléenne avec des valeurs de vérité **[33].**

Tableau 5: Les différents éléments assurant la confiance

4.2.3. Gestion de la confiance

La gestion de confiance est définie comme étant l'activité de création de systèmes et de méthodes pour permettre aux utilisateurs de faire des évaluations et de prendre des décisions concernant la fiabilité des opérations, et de leur permettre ainsi qu'aux propriétaires du système d'augmenter et représenter correctement la fiabilité de leur système **[34]**.

4.2.4. Importance de la confiance

La confiance et la sécurité sont étroitement liées et les processus de décision de la confiance les traitent de manière identique. Donc, les problèmes de confiance ne se poseront plus si une transaction entreprise s'est faite en toute sécurité et seront présents dans le cas contraire. C'est un pilier fondateur visant à assurer la sécurité du système et améliorer la QOS en assurant une composition optimale de services, à garantir d'autres critères de sécurité, tel que le contrôle d'accès et l'autorisation. **[27].**

4.2.5. Modèles de gestion de confiance

En étudiant l'état de l'art sur ces modèles, nous distinguons trois familles:

a. Confiance à partir de Credentials (Certification) :

Elle repose sur la mise en place d'une ou plusieurs politiques de sécurité et d'un système de certificats : les nœuds utilisent la vérification des certificats pour établir un lien de confiance avec d'autres nœuds **[39]** : Ainsi, un nœud a confiance en un autre si ce dernier à un certificat valide.

b. Confiance à partir de réputation et de recommandation

La gestion de la confiance repose sur un modèle de réputation et/ou de recommandation. De tels systèmes fournissent un mécanisme pour lequel un nœud DS peut

évaluer la confiance qu'il porte au FS. Chaque nœud établit ainsi des relations de confiance avec d'autres et affecte des valeurs de confiance **[42]** qui sont fonction d'une combinaison entre la réputation globale du nœud et l'évaluation de sa perception (son expérience).

c. Confiance à partir d'un réseau social

Les relations sociales sont utilisées pour calculer les valeurs de réputation et de recommandation pour chaque nœud. De tels systèmes analysent le réseau social qui représente les relations existantes dans chaque communauté dans le but de tirer des conclusions sur les niveaux de confiance à accorder aux autres nœuds. Ils reposent sur des mécanismes de réputation, de crédibilité, d'honnêteté et procédés de recommandations **[39].**

4.2.6. Attaques sur la confiance

Les plus courantes attaques exécutées par des entités malicieuses sont :

- **Self-Promoting :** Un nœud malveillant peut accroître son importance de manière à être sélectionné en tant que FS, mais peut fournir des mauvais services [**33**].
- **Bad-Mouthing :** Un nœud malveillant peut ruiner la réputation d'un nœud honnête de façon à diminuer la probabilité d'être sélectionné pour la réalisation de services **[33].**
- **Ballot-Stuffng :** Un nœud malveillant peut accroître la confiance d'un autre nœud malveillant de manière à être sélectionné en tant que FS [**33].**
- **On-Off :** Un nœud malveillant effectue un mauvais service afin d'éviter d'être étiqueté comme un nœud de confiance faible et de ne pas être sélectionné en tant que FS, ainsi que de ne pas pouvoir effectuer efficacement des attaques de type "Bad-Mouthing" et "Ballot-Stuffng" **[33].**
- **White-Washing :** Un nœud malveillant peut disparaitre puis rejoindre l'application pour laver sa mauvaise réputation **[38].**
- **Discriminatory :** Un nœud malveillant peut mener une attaque discriminante sur les nœuds non-amis ou sans beaucoup d'amis, cette attitude est similaire à celle des humains qui préfèrent interagir avec leurs Amis plutôt que des inconnus [**19].**
- **Déni de service (DS) :** Un nœud malveillant agit dans l'intérêt de rendre un autre nœud incapable d'assurer un service **[36].**

4.2.7. Taxonomie des modèles de confiance dans l'IoT

Nous avons défini les critères d'analyse et de comparaison des différents modèles de gestion de confiance proposés dans la littérature, puis nous les avons classés pour finir par une synthèse dans laquelle nous avons exposé leurs avantages et leurs inconvénients,

4.2.7.1. Critères de comparaison des solutions

Nous fonderons notre comparaison sur un ensemble de critères qui sont:

- **Résistance aux attaques**

Tout objet FS ou DS veut être sélectionné pour fournir un service à but lucratif s'il est FS ou recevoir des services des meilleurs FSs s'il est DS. Un FS malveillant agit dans le but d'être sélectionné pour la réalisation d'un service même si la QOS qu'il peut fournir est inférieure à celle assurée par d'autres objets. Les préoccupations sont les attaques pouvant perturber le système de gestion de confiance et ainsi causer une perte de précision dans l'évaluation de confiance.

- **Consommation énergétique**

La majorité des dispositifs du SIoT sont limités en termes de mémoire et des capacités énergétiques **[37].** Alors, la gestion de confiance doit être à moindre consommation d'énergie.

- **Évolutivité (Scalabilité)**

C'est une caractéristique essentielle d'un modèle de confiance dans SIoT. Elle permet d'assurer un correct fonctionnement lors de changement dynamique de la taille du réseau **[35]**.

- **Précision dans le calcul de la confiance**

Représente le degré de similarité entre le score de confiance d'un nœud calculé par le système de gestion de confiance avec la confiance effective qui doit être attribuée au nœud.

- **Monitorage**

Représente la capacité de suivi du comportement de tout objet, ce qui permet d'utiliser l'historique de ses agissements pour calculer avec précision son score de confiance.

4.2.7.2. Modèles de confiance dans l'IoT

Les modèles de confiance étudiés sont classés en deux branches :

a. Modèles de confiance distribués :

❖ **Mécanisme de gestion de la confiance:**

Gu Lize et al. **[44]** ayant considéré que chaque nœud peut-être DS ou FS. Ils ont proposé un mécanisme de gestion de confiance (sur les 3 couches composant l'architecture: Couche physique, réseau et application) qui se déroule conformément aux étapes décrites dans la figure14 dans le but d'assurer des services plus qualifiés et adéquats.

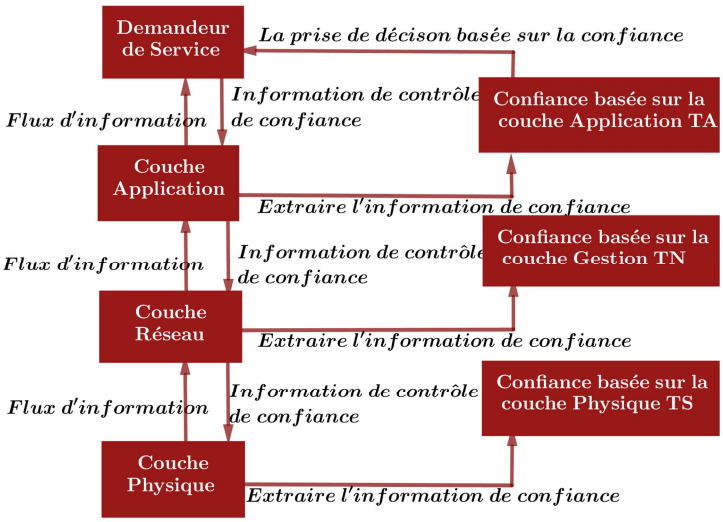

Figure 13: Processus de gestion de la confiance

Première étape : Un nœud DS émet une requête que le mécanisme de gestion de confiance traduit en informations de contrôle contenant le service requis, les facteurs de contrôle ou attributs qui serviront au calcul de la confiance et le contexte qui sont envoyés aux trois couches constituant l'architecture de l'IoT ;

Deuxième étape : À la réception de ces informations, les nœuds de contrôle de chaque couche collectent les informations de confiance de plusieurs objets et évaluent le niveau de confiance de chacun d'eux afin d'élire l'ensemble de nœuds jugés aptes à réaliser le service requis ;

Troisième étape : ce résultat représentant la confiance de la couche est transmis à la couche supérieure pour être combiné avec la confiance des autres couches afin de sélectionner le meilleur FS destiné à réaliser la requête du DS.

On outre, le mécanisme proposé permet d'assurer le contrôle d'accès en se basant sur la confiance attribuée à un utilisateur sachant qu'il appartient au système IoT.

- **Discussion et critiques** : Ce modèle permet :
  - Une bonne évolutivité grâce à son architecture distribuée et la sensibilité au contexte impliqué dans le calcul de confiance ;
  - Une grande consommation d'énergie due à l'overhead en termes de messages échangés lors du calcul de la confiance et la prise de décision ;
  - Le monitorage n'est pas assuré compte tenu du mécanisme distribuée ;
  - L'absence de sauvegarde des évaluations de confiance des nœuds ;
  - L'ignorance de la résistance aux attaques engendre une grande perte de précision dans le calcul de la confiance.

❖ Gestion de services basée sur la confiance pour les systèmes SIoT :

Ing-Ray Chen et al.**[51]** ont présenté un modèle autonome, adaptif et distribué de gestion de confiance pour SIoT basé sur 3 paramètres: **l'honnêteté**, **la coopérativité** et **l'intérêt à la *communauté*** suffisants pour la résistance aux attaques et le calcul de la confiance qui est effectué par des processus distribués et implémentés dans chaque objet, utilisant des sommes pondérées avec des coefficients dynamiques prenant en compte les expériences subjectives ainsi que les recommandations d'autres objets, tout en accordant un poids supérieur aux évaluations les plus récentes.

La propagation et la mise à jour de la confiance s'effectuent après chaque interaction. Les évaluations de confiance des objets et celles attribuées à d'autres nœuds sont échangées, sachant qu'un objet ne garde en mémoire que celles des objets avec lesquels il a interagit.

- **Discussion et critiques** : Ce modèle permet :
  - Garantir une bonne évolutivité et une adaptation au contexte grâce à l'utilisation de coefficients dynamiques dans les sommes pondérées ;
  - Une modeste consommation en énergie due à l'utilisation de simples calculs (sommes pondérées) combinés avec une mise à jour de confiance orientée événement, ce qui diminue le nombre de messages échangés entre les nœuds;
  - Le monitorage n'est pas assuré, étant donné l'absence d'infrastructure centralisée pour la sauvegarde de la réputation de chaque objet ;
  - Il ne résiste pas aux attaques de type "White-Washing", mais résiste aux attaques de type "Self Promoting" , "Bad Mouthing ", " Ballot Stuffing" et "On-Off " grâce à l'utilisation de l'honnêteté comme paramètre lors du calcul de la confiance, de plus les attaques de type "Discriminatory" sont évitées en se basant sur les propriétés de coopérativité et d'intérêt à la communauté pour calculer la confiance.

❖ Une approche floue du contrôle d'accès basé sur la confiance:

P.N. Mahalle et al. **[50]** ont proposé un modèle à base de confiance afin de garantir le contrôle d'accès en utilisant un Framework chargé du calcul de la réputation de chaque nœud et lui faire correspondre les permissions associées. Le mécanisme se déroule en 3 étapes:

- **Première étape** : Consiste à collecter les informations suivantes sur chaque objet du réseau:
  - L'expérience (EX) : $\mathbf{EX} = \frac{Nombre\ d'intéractions\ réussies}{Nombre\ total\ d'intéractions}$ et prend une des valeurs linguistiques suivantes (**Mauvaise, Moyenne, Bonne**).

- Connaissances (KN) : Calculées à partir des évaluations directes et indirectes prenant les valeurs linguistiques (**Insuffisante, Assez, Complète**).
- Recommandations (RC) : Obtenues en agrégeant les avis des autres objets prenant les valeurs linguistiques suivantes (**Négative, Neutre, Élevée**).

Les valeurs linguistiques précédentes sont traduites en valeurs floues appartenant à l'intervalle [-1,1] en plus de l'introduction de fonction d'appartenance prenant des valeurs dans l'intervalle [0,1] et permettant de représenter le degré de vérité d'une valeur linguistique.

Expérienc	Connaissan	Recommandation	Plages de novaux	Les valeurs floues
Mauvaise	Insuffisante	Négative	Au-dessous de $-0,5$	$-1;\ -1; -0,5; -0,1$
Moyenne	Assez	Neutre	$-0,1; 0,25$	$-0,25;\ -0,1; 0,25; 0,5$
Bonne	Complète	Élevée	Au-dessous de $0,5$	$0,25;\ 0,5; 1; 1$

Tableau 6: Division des variables linguistiques en valeurs linguistiques

- **Deuxième étape** : Elle consiste à utiliser des règles d'inférences définies par les auteurs telle que chaque règle prend en entrées les valeurs linguistiques de EX, KN et RC afin de produire en sortie une valeur de confiance qui appartient à un des ensembles flous prenant les valeurs linguistiques (bonne, moyenne ou mauvaise), et sera considérée comme l'évaluation de confiance associée au nœud.
- **Troisième étape** : Elle consiste à faire un mappage des évaluations de confiance sur les autorisations d'accès, cela en répartissant les droits d'accès dans des sous-ensembles réunis au sein d'un seul ensemble ayant une cardinalité égale au nombre de valeurs linguistiques en sortie du système d'inférence utilisé, ce qui permet de fournir l'accès aux ressources ou aux appareils avec le principe du moindre privilège.

- **Discussion et critiques** : Ce modèle permet :
  - Une bonne évolutivité grâce à son architecture distribuée.
  - L'ignorance de la résistance aux attaques engendre une grande perte de précision dans le calcul de la confiance.
  - Forte consommation d'énergie due à l'overhead en termes de messages provoqué par le processus de collecte d'informations sur chaque nœud.
  - Aucun mécanisme de monitorage n'a été défini.

❖ Gestion de la confiance évolutive, adaptative et durable pour les systèmes d'IoT fondés sur des communautés d'intérêts :

Fenye Bao et al. **[49]** ont proposé un modèle de gestion de confiance pour le SIoT utilisant le QOS et les métriques spécifiques au SIoT (honnêteté, coopérativité, intérêt à la communauté) comme composantes de la confiance.

Chaque nœud sauvegarde en mémoire les valeurs de confiance attribuées aux nœuds avec lesquels il a interagit. Ses sauvegardes serviront à assembler les observations subjectives en confiance directe en utilisant des sommes pondérées et aussi les recommandations en confiance indirecte en utilisant la similarité sociale ainsi que les sommes pondérées dynamiques pour ajuster dynamiquement le poids associé à la confiance directe et indirecte, ce qui permet la maximisation des performances et la sensibilité au contexte.

La mise à jour de la confiance s'effectue après chaque interaction en privilégiant les interactions récentes en leur attribuant un poids plus fort dans la somme pondérée, ce qui donne au modèle la possibilité d'adaptation au dynamisme du SIoT tout en évitant la discrimination des nouveaux nœuds.

- **Discussion et critiques :** Ce modèle permet :
  - De garantir une évolutivité élevée grâce à son architecture distribuée ;
  - Ne garantit pas la mobilité à cause de l'absence d'infrastructure centrale dédiée à la sauvegarde des valeurs de confiance de tous les nœuds du réseau ;
  - Une économie d'énergie au niveau des nœuds du réseau à cause de l'utilisation de sommes pondérées dans le calcul de la confiance et la nature orienté événement de la mise à jour de ces valeurs ;
  - D'assurer la résistance aux attaques en combinant les trois paramètres honnêteté, coopérativité et intérêt à la communauté pour détecter les nœuds malicieux ce qui garantit une bonne précision du calcul de la confiance.

b. Modèles hiérarchiques

❖ Modèle de confiance fondé sur la réputation et la garantie pour SIoT;

Hannan Xiao et al**. [48**] ont proposé un modèle hiérarchique de gestion de confiance pour le SIoT utilisant deux paramètres :

- **La réputation** : Employée pour déterminer le niveau de fiabilité d'un nœud dans l'accomplissement d'un service donné ;
- **Le crédit:** fait office de commission à céder pour obtenir un service digne ou de caution dans le cas contraire afin de détecter les nœuds malicieux ;

La gouvernance du système est assurée au niveau de la couche application en utilisant une chaine de serveur de deux types différents :

- **Le premier type** : représente les serveurs de réputation chargés du calcul, la mise à jour et la propagation des valeurs de réputation des nœuds du réseau ;

- **Le second type** : représente les serveurs de bases de données prenant en charge la découverte de services ainsi que le choix des meilleurs chemins pour y accéder. Ex :
  - ✓ **Envoi d'une requête** : Un DS l'envoie au gestionnaire de confiance à travers son point d'accès dans laquelle il indique le service, la commission qu'il est prêt à payer en échange de sa réalisation, ainsi que le forfait que le FS lui cédera si sa réalisation est jugée de mauvaise qualité.
  - ✓ L**a réception de la requête** : le gestionnaire de confiance interroge le serveur de bases de données pour avoir l'ensemble des objets pouvant fournir ce service, puis sélectionne parmi eux l'objet qui a la réputation la plus élevée.
  - ✓ **Fourniture de service** : L'objet sélectionné fournit le service au DS, qui procède après à l'évaluation du QOS reçu qui sera envoyée sous forme de rapport au serveur de réputation.

Notes : En cas de réalisation d'un bon service, le DS paye la commission convenue au FS, et sinon le FS cède un forfait qui est une sorte d'indemnisation.

Si un objet malicieux fournit de faux rapports à propos d'un objet, alors ce dernier conteste auprès du gestionnaire de confiance afin que l'émetteur du rapport soit sanctionné.

- **Discussion et critiques :** Ce modèle permet :
  - Assurer le monitorage grâce à la centralisation de la sauvegarde des évaluations de chaque nœud.
  - La consommation de ressources, la charge de calcul est transmise à la couche application qui est très riche en ressources.
  - Le critère d'évolutivité n'est pas garanti ;
  - La résistance aux attaques présente de nombreuses lacunes relevant de :
    - ✓ Absence de gestion, distribution de crédits entre les nœuds et la vérification QOS ;
    - ✓ La détection des nœuds malicieux effectuée par les nœuds.

Conséquence : Puisque ce mécanisme n'est pas résistant aux attaques, ça engendre une perte considérable de précision dans le calcul de la confiance.

❖ Gestion de la Confiance dans le SIoT :

Michèle Nitti et al. **[47]** ont proposé deux modèles de gestion de confiance pour le SIoT, le 1er est subjectif, le 2^{e} objectif. Les deux utilisent la QOS ainsi que les caractéristiques du SIoT pour la composition de la confiance.

- **Gestion de confiance subjective** : chaque nœud A calcule la valeur de confiance qu'il attribue à un nœud B avec lequel il a interagit, puis sauvegarde cette valeur, en plus d'un bilan lié à la QOS assuré pour pouvoir le diffuser à la demande d'un autre nœud C ayant un lien social avec A. Le nœud C utilise la recommandation apportée par A, (sa centralité et son expérience directe avec A comme facteur dans une somme pondérée) afin d'évaluer B. La mise à jour des valeurs de confiance est orientée événement, telle que chaque interaction entre 2 nœuds est suivie d'une évaluation mutuelle des valeurs de confiance.
- **Gestion de confiance objective** : C'est une structure centralisée où les informations liées à la QOS assurées par chaque nœud sont envoyées à une entité centrale, qui utilise une DHT afin de maintenir la réputation globale de chaque nœud. Un nœud ayant a besoin de la valeur de confiance d'un autre, émet une requête à la DHT qui cherchera cette valeur dans la base de données et la lui retournera.

La mise à jour des informations sauvegardées dans la DHT est effectuée après chaque interaction, tel que chaque nœud A recevant un service de la part du nœud B, émet un rapport décrivant le service fourni.

- **Discussion et critiques :**

L'approche subjective	**L'approche objectif**
– Permet l'économie d'énergie ;	– Permet de retirer le fardeau de calcul aux nœuds du réseau, ce qui implique une moindre consommation de ressources ;
– Garantit l'évolutivité par sa nature distribuée ;	– Assure l'évolutivité du fait qu'il est inspiré des technologies du réseau P2P ;
– Le monitorage n'est pas possible à cause de l'absence de stockage centralisé des valeurs de confiance des nœuds du réseau ;	– Assure le monitorage grâce au stockage des informations de confiance de chaque nœud du réseau et leur disponibilité ;
– Résiste aux attaques de type "Self-Promoting", "Bad-Mouthing" et "Ballot-Stuffing", grâce à l'utilisation de la crédibilité comme poids pour les recommandations, mais non aux attaques de types "On-Off" et "white-waching", ce qui cause une perte de précision lors du calcul de la confiance.	– Ne résiste pas aux attaques de type "On-Off", ce qui cause une perte de précision lors du calcul de la confiance.

❖ Conception d'un système de gestion de la confiance pour l'IoT: une approche contextuelle et multiservice :

Ben saied et al. **[46]** ont proposé un modèle de gestion de confiance hiérarchique à base de réputation afin garantir une composition de services précise, optimale et sensible au contexte pour l'IoT. Le processus se déroule selon cinq phases :

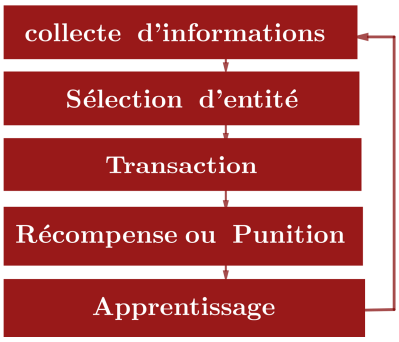

Figure 14: Processus de gestion de la confiance

Première phase : Collection des rapports émis par des objets ayant reçu un service. Chacun contient le service fournit, son temps d'exécution, le score affecté par le témoin à l'objet fournisseur du service ainsi que son état courant (vieillissement, capacité en ressources).

Deuxième phase : Sélection des meilleurs proxys en privilégiant les nœuds ayant fourni le même service dans des conditions équivalentes tout en favorisant ses services les plus récents dans la sélection.

Troisième et Quatrième phase : Fourniture du service et sera suivie de l'évaluation des objets FSs.

Cinquième Phase : est subdivisée en deux étapes :

- ✓ **Première étape**: est la mise à jour de la qualité des recommandations des nœuds ayant fourni les rapports, en effectuant la comparaison entre elles ;
- ✓ **Deuxième étape** : est la mise à jour des réputations des nœuds FS en calculant la somme des scores attribués par les nœuds DS avec leurs qualités de recommandations.

- **Discussion et critiques :** Ce modèle permet :
  - Le monitorage et une faible consommation d'énergie grâce à sa nature hiérarchique ;
  - Ne garantit pas l'évolutivité et ne résiste pas aux attaques de type " On-Off " puisque le mécanisme de punition implémenté ne prend pas en compte le comportement des nœuds à long terme, causant une perte de précision dans le calcul de la confiance.

❖ Modèle de gestion de confiance à base de crédit et d'honnêteté

- **Modèle physique**

Il repose sur une architecture hiérarchique composée de serveurs de communautés entre lesquels il existe des relations de confiance dans le but de garantir son évolutivité :

- Un objet DS envoi une requête à travers son point d'accès qui sera transmise après au serveur de communauté à laquelle il appartient et qui assure la gestion de confiance et la découverte de services assurés par les objets qu'il supervise.
- À la réception de la requête, un objet FS est sélectionné sur la base de son score de confiance.

Si la requête ne peut être satisfaite par un objet de la même communauté (interaction intra-communauté), le serveur la transmet à une autre communauté avec laquelle il entretient une relation de confiance (interaction inter-communauté), d'où l'utilité de la répartition du système de gestion de confiance sur plusieurs serveurs physiques afin de créer une seule entité de gestion virtuelle, à savoir le gestionnaire de confiance.

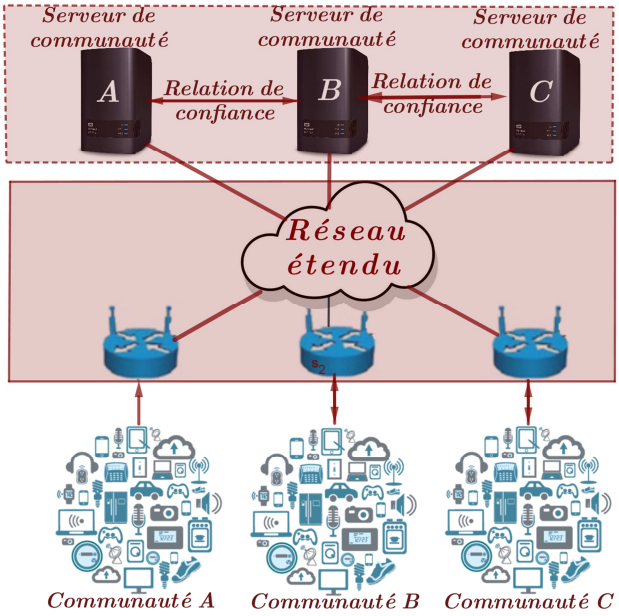

Figure 15: Modèle physique du système de gestion de confiance

- **Fonctionnement du gestionnaire de confiance :** Ce modèle de confiance procède conformément aux cinq phases qui sont **[45]:**

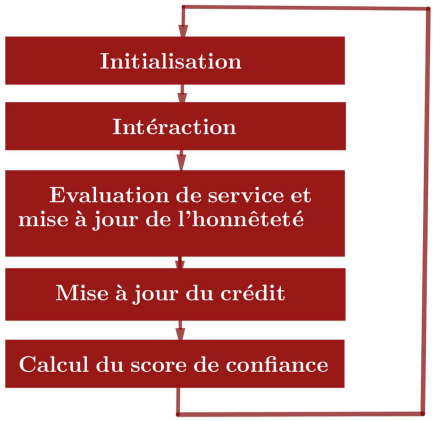

Figure 16: Les différentes phases du modèle de confiance à base de crédit et d'honnêteté

✓ Initialisation :

Après le déploiement du réseau, le gestionnaire de confiance initialise la réputation, le score de confiance et le crédit affecté à chaque objet sachant que tout est neutre au début. Le calcul du crédit initial affecté à un objet est défini par la formule :

$$C_i = C_d \times (1 + \frac{1}{C_{ap}}) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

La capacité (C_{ap}) d'un objet est calculée en faisant intervenir : l'énergie, l'espace de stockage et la puissance de calcul, ceci conformément à la formule suivante :

$$C_{ap} = \rho . E + \lambda . M + \mu . P \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

✓ Interaction

Un DS génère une requête contenant : le service, le seuil minimum de confiance que doit avoir le FS, la commission qu'il est prêt à payer en contrepartie ainsi que le forfait que le FS lui cèdera dans le cas de la réalisation d'un mauvais service. À sa réception, l'objet satisfaisant les exigences du service demandé sera sélectionné dans la même communauté, sinon, la requête sera retransmise au serveur d'une autre communauté avec laquelle il entretient une relation de confiance. Quand un objet reçoit une requête peut soit : Accepter la réalisation du service et le fournir au DS qui l'évaluera, ou la refuser et répondre par message au gestionnaire qui sélectionnera un autre FS.

✓ Évaluation de service et mise à jour de l'honnêteté

À la fin de chaque interaction, le DS envoie au gestionnaire de confiance un rapport d'appréciation (**Taux de satisfaction**) du FS qui sera enregistré pour déduire l'honnêteté de l'émetteur du rapport, et sera utilisé ultérieurement pour le calcul de la confiance du FS.

La formule(3) permet de calculer la similarité du taux de satisfaction, en comparant le taux de satisfaction émis par le DS (d) sur le FS (f) ayant fourni un service (s) avec tous les taux de satisfaction émis par d'autres objets sur le FS (f) concernant le même service ou un service dont la complexité est semblable à (s'). La valeur calculée appartient à] 0, 1] qui représente le pourcentage d'honnêteté de l'objet DS dans le jugement du FS.

$$St_f^d(s) = \frac{T_f^d(s)}{\sum_{i=1}^{N} \frac{T_f^i(s').H^i}{N}} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

L'utilisation de la similarité de taux de satisfaction permet d'éviter les attaques de types Bad-mouthing, Good-mouthing et Ballot-stuffng. Un objet ne peut pas mentir sur la QOS effective d'un autre objet, car dans le cas contraire il est directement sanctionné en diminuant son honnêteté et finira par être exclu du réseau dans le cas de récidive.

$$H^d(t) = (1 - \sigma) \times H^d(\Delta t) + \sigma [St_f^d(s)] \ldots\ldots\ldots (4)$$

✓ Mise à jour du crédit :

Le processus de mise à jour du crédit est un facteur déterminant dans le calcul de la confiance. Il est exécuté après chaque interaction afin que le FS soit récompensé ou punit.

- **Récompense** : Si le service fourni est conforme à ce qui a été convenu dans la requête du DS, le serveur de communauté déduit un nombre de crédit du DS selon la formule (5) et l'additionne au crédit du FS selon la formule (8) :

$$C^d(t) = C^d(\Delta t) - Co_f^d(s) \text{.....(5)}$$
$$C^f(t) = C^f(\Delta t) + Co_f^d(s) \text{ .....(6)}$$

- **Punition :** Un Objet est puni dans les deux cas suivants :

1. **Fourniture d'un mauvais service** : Si un FS ne satisfait pas les conditions convenues dans la prestation de service offert, il est sanctionné en diminuant son crédit conformément à la formule (7). Le DS reçoit un forfait qui représente une indemnisation et additionnée à son crédit conformément à la formule (8) :

$$C^f(t) = C^f(\Delta t) - \varphi F_f^d(s) \text{ Tel que : } \varphi = \frac{Bs^d}{Ts^d} \text{ ..... (7)}$$
$$C^d(t) = C^d(\Delta t) - F_f^d(s) \text{ ..... (8)}$$

Note : La punition est toujours plus sévère que la récompense afin d'éviter les attaques de type On-Off.

2. **Oisiveté :** Un objet est oisif s'il ne fournit pas de service pendant une certaine durée parce qu'il est soit malveillant ou bien défectueux.

Dans le but d'encourager les objets bienveillants à interagir et détecter les nœuds malveillants ou défectueux, le gestionnaire de confiance diminue le crédit des objets n'ayant pas interagi pendant un temps fixé (t_0) calculé en ôtant le temps de la dernière interaction (t_i) de l'instant actuel (t): alors le nœud est puni conformément à la formule 9 si le temps de l'oisiveté est supérieur à (t_0) :

$$Si\ t - t_i > t_0 \quad Alors \quad C^f(t) = C^f(t) - \omega C^f(t) \text{.....(9)}$$

Note : L'attaque de DOS est exclue dans ce modèle car tout service à un coût et un nœud voulant attaquer un autre doit d'être un bon FS afin de gagner assez de crédit pour demander des services, ainsi un nœud malveillant devra épuiser ses ressources pour pouvoir attaquer un autre objet contraire aux modèles ne mettant pas de crédits car un nœud malicieux peut épuiser les ressources d'un ou plusieurs objets en effectuant ce genre d'attaque.

✓ Mise à jour du score de la confiance

Après que l'honnêteté et le crédit sont mis à jour, le gestionnaire de confiance calcule le score de confiance des deux nœuds ayant interagi en utilisant la formule suivante :

$$CF_0(t) = (1 - \gamma)H^0(t) + \gamma C^0(t) \text{.....(10)}$$

Param.	Signification
C_i	Crédit initial affecté à un objet i ;
C_d	Valeur de crédit minimum nécessaire pour une interaction.
C_{ap}	Capacité d'un objet (relative aux ressources).
ρ	Facteur permettant d'ajuster le poids de l'énergie d'un objet par rapport aux autres ressources.
E	Énergie d'un objet.
λ	Facteur permettant d'ajuster le poids de la capacité de stockage d'un objet par rapport aux autres
M	Capacité de stockage d'un objet
μ	Facteur permettant d'ajuster le poids de la puissance de calcul d'un objet par rapport aux autres ressources.
P	Puissance de calcul d'un objet.
s	Service.
s'	Service similaire au service S (soit même service soit complexité équivalente).
$St_f^d(s)$	Similarité du taux de satisfaction émis par l'objet d sur un service S fourni par un objet f avec les autres taux émis à propos du même service(ou un service similaire) fournis par le même objet f.
$T_f^d(s)$	Taux de satisfaction émis par l'objet d sur un service S fourni par un objet f.
$T_f^i(s')$	Taux de satisfaction émis par l'objet i sur un service s' fourni par un objet f.
N	Nombre total d'appréciations émises sur le service S ou un service similaire en termes de complexité par un objet quelconque.
$H^d(t)$	Honnêteté d'un objet d à l'instant t.
σ	facteur permettant d'ajuster le poids de l'évaluation récent de l'honnête par rapport à l'ancienne valeur.
$C^d(t)$	Crédit associé à l'objet d demandeur de service a l'instant t.
$Co_f^d(s)$	Instant de la dernière interaction.
$C^f(t)$	Commission négocié par le DS d avec un FS f concernant le service S.
φ	Crédit associé à l'objet f fournisseur de service a l'instant t.
Bs^d	Facteur représentant la fréquence de mauvais service e
Ts^d	Nombre de bons services fournis par d.
F_f^d	Nombre total de services fournis par d.
t_0	Forfait négocié par le DS (d) avec un FS (f) concernant le service ;
ω	temps d'oisiveté toléré ;
$CF_0(t)$	Score de confiance d'un objet o à l'instant t ;
γ	Facteur permettant d'ajuster le poids de l'honnêteté dans le calcul de la confiance ;
$H^0(t)$	Évaluation de l'honnêteté d'un objet o à l' instant t.
$C^0(t)$	Évaluation du crédit d'un objet o à l' instant t.

Tableau 7: Les paramètres utilisés dans les formules de dérivation de la confiance

Chapitre 3 : Conception de SIoT

1. Introduction

Les principales contributions de cette section sont la proposition d'une architecture généralisée pour SIoT basée sur diverses architectures existantes. De plus, une explication ontologique et sémantique des composants du SIoT est délibérée ainsi que les différentes relations. La section est divisée en cinq paragraphes, comme indiqué ci-dessous:

Avec une proposition d'une introduction au nouveau paradigme du SIoT au début, puis d'une présentation d'une architecture généralisée de SIoT et une architecture conceptionnelle avec les deux parties qui la composent: Côté client et côté serveur, et enfin conclusion du chapitre par fourniture d'une explication sémantique d'un réseau SIoT.

2. Architecture SIoT

Dans ce paragraphe, on illustre certaines architectures prédominantes pour SIoT. En outre, nous proposons une architecture généralisée incorporant les caractéristiques clés de celles qui prévalent :

- Une architecture à 3 couches pour l'IoT est définie et qui sont : couche de détection, de communication et d'application **[83]**.
- 3 différentes architectures sont définies pour le serveur SIoT, la passerelle et l'objet **[77]**.
- L'architecture de la plateforme de réseau social est conçue pour définir un modèle architectural pour les réseaux SIoT. En outre, il est proposé une architecture à 3 couches côté client et côté serveur **[84]**.
- Il est défini une architecture basée sur un stockage distribué pour SIoT dans **[41]**.
- Il est défini un cadre et une architecture pour le client et le serveur SIoT dans **[47]**.

Dans ce chapitre, nous fournissons un aperçu du système SIoT. La section 2.1 présente le cadre général du système et la 2.2 une architecture conceptuelle.

2.1. Cadre général

Pour une meilleure compréhension **[76],** nous pouvons considérer les composants suivants (**Figure 17**) comme faisant partie de l'Internet social des objets :

- **Éléments intelligents et utilisateur** : Conformément aux exigences de base de l'IoT, chaque objet physique ou virtuel est capable de capturer et de transmettre des données.

Ainsi, dans SIoT, l'homme et les appareils peuvent participer de manière égale à la publication des données qui peuvent être des données de profilage ou une réponse à leurs requêtes.

- **Système intelligent** : responsable du traitement des requêtes, de la gestion des applications, de la découverte de services, des recommandations, de la gestion des données et du contexte.
- **Interface** : Moyen d'interaction entre un utilisateur ou un dispositif d'un côté et le système intelligent d'un autre, c'est-à-dire, elle assure la connectivité et le transfert de données en temps réel. Les interfaces intelligentes peuvent également communiquer entre elles pour partager des données.
- I**nternet** : Moyen de communication entre l'utilisateur et les appareils. En outre, cela aide à interagir entre les appareils.

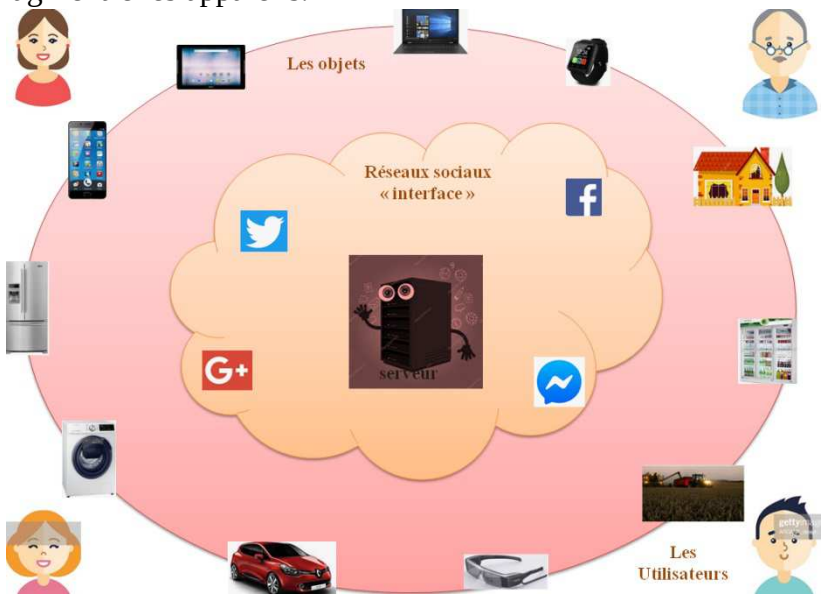

Figure 17: Composants du SIoT

2.2. L'architecture conceptuelle

2.2.1. Les différentes couches d'un SIoT et leurs fonctionnements

Sur la base des services et des composants définis ci-dessus, nous proposons une architecture généralisée à quatre couches pour SIoT (comprenant les couches objet, communication, gestion et application), illustrée à la **Figure 18.**

Le fonctionnement des couches respectives est décrit comme suit:

a. **Couche d'objets:** La couche de base où sont présents les objets IoT et les dispositifs équipés de capteurs. Ces appareils peuvent collaborer et communiquer les uns avec les autres via des réseaux de capteurs locaux.

b. **Couche de communication:** Définit les technologies et les protocoles de communication entre les périphériques et les réseaux du SIoT.

c. **Couche de gestion SIoT:** Définit la plate-forme requise pour gérer les services SIoT. Les composants essentiels composent les services SIoT.

d. **Couche d'application:** Elle assure l'interfaçage avec diverses applications SIoT et inclut également des APIs.

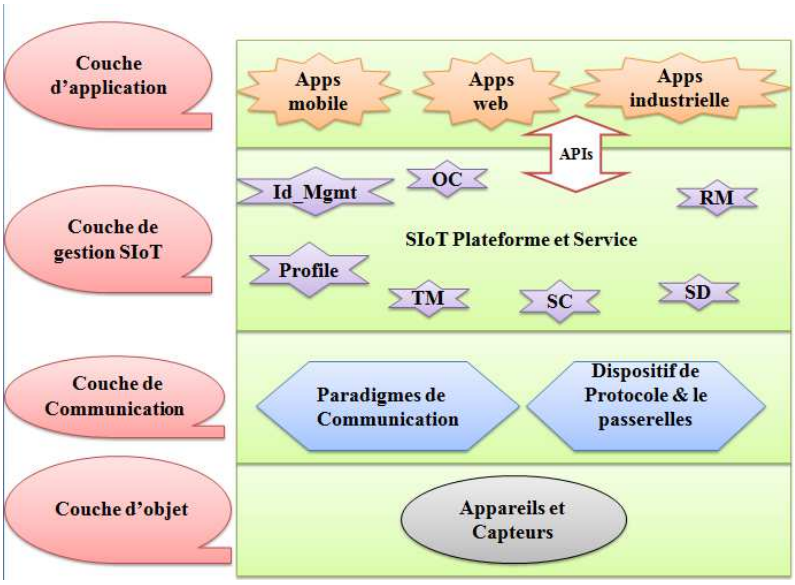

Figure 18: Architecture de référence SIoT

2.2.2. Décomposition de l'architecture SIoT

Dans ce paragraphe, nous proposons l'architecture d'un Internet des objets social. Nous avons classé l'architecture en deux parties, côté serveur et côté client :

2.2.2.1. Côté client

Côté client (**Figure 19**) composé de quatre couches :

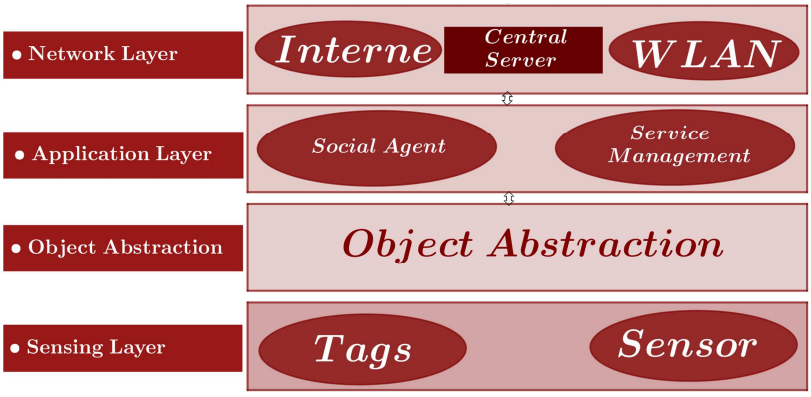

Figure 19: Architecture côté Client

- **Couche de détection de données** : est l'endroit où se trouvent les objets physiques et virtuels. où sont détectées les données réelles à l'aide de capteurs et où peuvent

communiquer les uns avec les autres via Internet. Différentes étiquettes (RFID) sont utilisées à des fins d'identification **[75]**.

- **Couche d'abstraction d'objets:** Elle est nécessaire pour harmoniser la communication de différents périphériques en convertissant le format de données dépendant du périphérique en un format commun afin qu'elles puissent facilement être transmises sur le réseau.
- **Couche d'application**: Ici, l'agent social est engagé pour communiquer entre les objets et le serveur SIoT afin de mettre à jour le profil, les amitiés. En outre, il transporte une requête du réseau social. La gestion de service est l'interface des humains permettant de contrôler le comportement de l'objet lors de la communication au sein d'un réseau social.
- **Couche réseau** : La couche de détection des données collectées par les capteurs est envoyée à la base de données ou au serveur central via différents réseaux: Internet, réseau WBAN (Wireless Body Area Network) et réseau local sans fil (WLAN). WLANS aide à la communication entre les appareils et Internet agit comme un moyen de communication entre l'utilisateur et les appareils.

2.2.2.2. Côté serveur

Côté serveur (**Figure 20**) composé de trois couches :

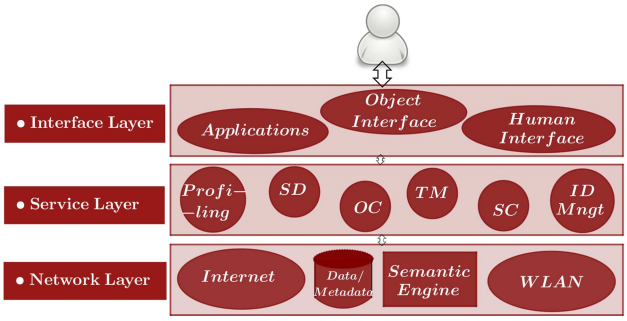

Figure 20: Architecture côté serveur

- **Couche réseau :** Le serveur SIoT **[74]** ne nécessite pas de couche de détection. Mais la couche réseau joue un rôle primordial côté serveur, car, elle permet de transférer des données de la base de données à l'interface utilisateur. Les ontologies sont stockées dans des bases de données distinctes. Il représente également la vision sémantique des activités sociales. Des ontologies et différentes vues sémantiques sont nécessaires pour fournir un cadre interprétable par machine permettant de représenter les opérations des dispositifs IoT. En outre, il est utile de représenter les attributs fonctionnels et non fonctionnels des périphériques.

- **Couche de service :** Contient les outils permettant d'implémenter les fonctionnalités de base de l'Internet social des objets :
  - **Gestion des identifiants (ID Mngt) :** Donne une identification universelle à l'objet.
  - **Profilage**: Donner des informations sur l'objet de manière automatique ou manuelle.
  - **Contrôle du propriétaire (OC):** Définit les activités que peut effectuer l'objet.
  - **Relationship management (RM)** : Établit et maintient des relations entre les objets. Des règles spécifiques sont définies pour la sélection de l'amitié parmi les objets ;
  - **Service Discovery (SD):** Décide du périphérique qui peut fournir un service requis.
  - **Composant de service (SC):** Permet l'interaction entre les objets. Elle consiste à récupérer des informations sur des requêtes ou à trouver le FS.
  - **Trustworthiness Management (TM):** Accorde ou refuse un service d'un dispositif.
- **Couche d'interface :** La 3^{e} couche où s'y trouve l'objet, l'homme et les services. C'est à travers elle que peut réagir l'utilisateur avec le serveur Web.

N	Composant	Son objectif
1	**ID management (ID Mgmt)**	Identifier de manière unique chaque objet du réseau ;
2	**Profiling**	Pour configurer des informations statiques et dynamiques sur les objets ;
3	**Owner control (OC)**	Pour gérer les schémas définis par le propriétaire pour le contrôle d'accès, les activités des objets et la création de relations ;
4	**Relationship management (RM)**	Établir et maintenir des relations entre les objets. Des règles spécifiques sont définies pour la sélection de l'amitié parmi les objets ;
5	**Service discovery (SD)**	Pour rechercher des objets fournissant les services requis à partir du réseau de relations sociales. Les objets imitent le comportement humain tout en recherchant des objets amis capables de leur fournir les services souhaités ;
6	**Service composition (SC)**	Permet aux objets de se traiter les uns les autres en utilisant différentes approches ;
7	**Trustworthiness management (TM)**	Responsable de l'évaluation de la confiance de différents objets pour le partage d'informations et la gestion des relations.

Tableau 8: Composants essentiels d'un réseau SIoT

3. Description sémantique de SIoT

Le concept de SIoT a été évoqué d'un point de vue sémantique dans ce paragraphe :

3.1. Description des objets:

Le nombre total d'objets comprend un total de 50 périphériques, dont 50 d'utilisateurs **:**

id_device	id_user	type d'appareil	marque_appareil	modèle d'appareil

Où :

id_device	Reference de l'appareil;
id_user	ID de propriétaire de l'appareil (nous indiquons la commune avec 0);
type d'appareil	Catégorie associée (code) à l'appareil; Les codes de type d'appareil sont expliqués
marque_appareil	allant de 1 à 12;
Modèle d'appareil	allant de 1 à 24.

3.2. Profil d'objets :

Pour chaque type d'appareil, le profil définit l'ensemble des services possibles offerts par chaque type ainsi que les applications possibles qui l'intéressent (i e : les applications qu'un objet pourrait demander). Les profils d'objets sont décrits sous la forme suivante:

type d'appareil	id_off_service	id_req_application

Où :

type d'appareil	Code associé à l'appareil;
id_off_service	liste des identifiants de service offerts (de 1 à 18);
id_req_application	liste des identifiants d'application (de 1 à 28).

3.3. Appareil privé:

Chaque utilisateur possède un certain nombre de périphériques, le tableau suivant indique le périphérique utilisé sur le réseau que l'utilisateur possède.

Dispositif	Mobilité	Possession (%)	Type d'appareil
Téléphone intelligent	Mobilité	91%	1
Voiture	Mobilité	55%	2
Tablette	Mobilité	40%	3
Pc	Statique	84%	4
Imprimante	Statique	53%	5
Thermomètre	Statique	35%	6

Tableau 9: Les périphériques utilisés dans le réseau

Les appareils mobiles sont transportés avec les utilisateurs lors de leurs déplacements, ainsi que les objets statiques sont laissés dans leur domicile.

Le fichier private_static_devices décrit les objets statique sous la forme :

Id_device	X	Y

Où :

id_device	Référence de l'appareil
X	X : coordonné de l'appareil
Y	Y : coordonné de l'appareil

Chaque utilisateur et ses appareils possèdent un état de mouvement et un état de repos. Le fichier private_mobile_devices décrit les informations sur les objets mobiles et les coordonnés des utilisateurs pendants l'état de repos sous la forme suivante :

timestamp start	timestamp stop	Id_user	X	Y

Où :

timestamp_start	Horodatage du début de l'état de ropos
timestamp_stop	Horodatage du début de la fin de repos
id_user	Identifiant d'utilisateur
X	x : coordonné de l'appareil
Y	y : coordonné de l'appareil

Les objets mobiles du même propriétaire ont la même position de l'utilisateur.

3.4. Matrices d'Adjacence :

Selon (SIoT), les nœuds établissent des liens sociaux et créent des réseaux sociaux. Nous proposons pour chaque relation et pour le réseau SIoT la matrice d'adjacence produite avec nos paramètres. Comme suit, nous décrivons les relations et leurs paramètres:

Les relations	Description	Paramètres
Relationship (POR)	Établie si les objets ont le même fabricant et construits à la même période;	Seuil de 2 à 2.5 km
Co-location object	Établie si les objets partagent leur emplacement ;	+ de 13 fois
Co-work object	Établi lorsque la nature du travail pour les objets est identique ;	
Ownership object	Établie si les objets appartiennent à la même personne;	Wi-Fi (≈ 400 m)
Social objects relationship	Établie lorsque la liaison entre les objets est fréquente ou récurrent ; N : nombre de réunions, T M : durée de réunions, T I : intervalle entre deux réunions	N = 3, T M = 1 minute, T I = 1 h.

Tableau 10: Les relations et leurs paramètres

4. Diagramme de SIoT

Il existe trois classes principales et la relation entre elles est décrite par les propriétés d'objet énumérées dans le **Tableau 9** :

Classes	Propriétaire	Objet	Composants SIoT
Sous-classes		Profile	Control Manager (CM)
		ID	Information Manager (IM)
			ID Manager (IDM)
			Profiling Manager (PM)
			Service Manager (SM)
			Friendship Manager (FM)
			Trust Manager (TM)
			Privacy and Security Manager (PSM)

Tableau 11: Les différentes classes composant l'ontologie

Les requêtes en langage sémantique permettent à l'utilisateur de poser des questions à partir du Web sémantique, à l'instar des phénomènes de requête-réponse avec des bases de données. De nombreuses langues sont disponibles pour interroger des données RDF / OWL.

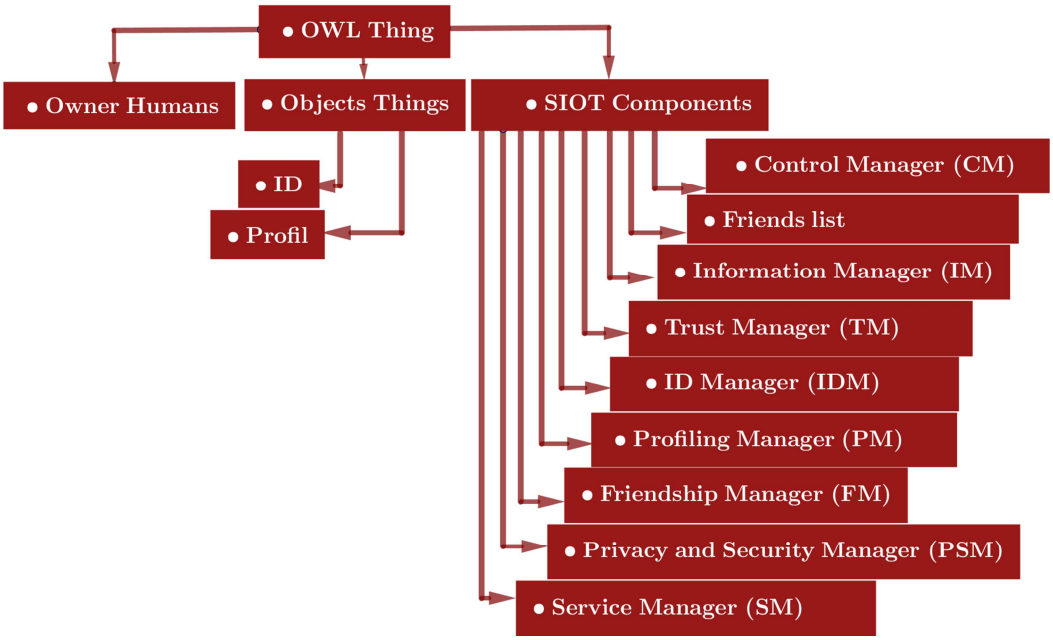

Figure 21: Diagramme de classes pour l'ontologie SIoT

5. Fonctionnement de SIoT

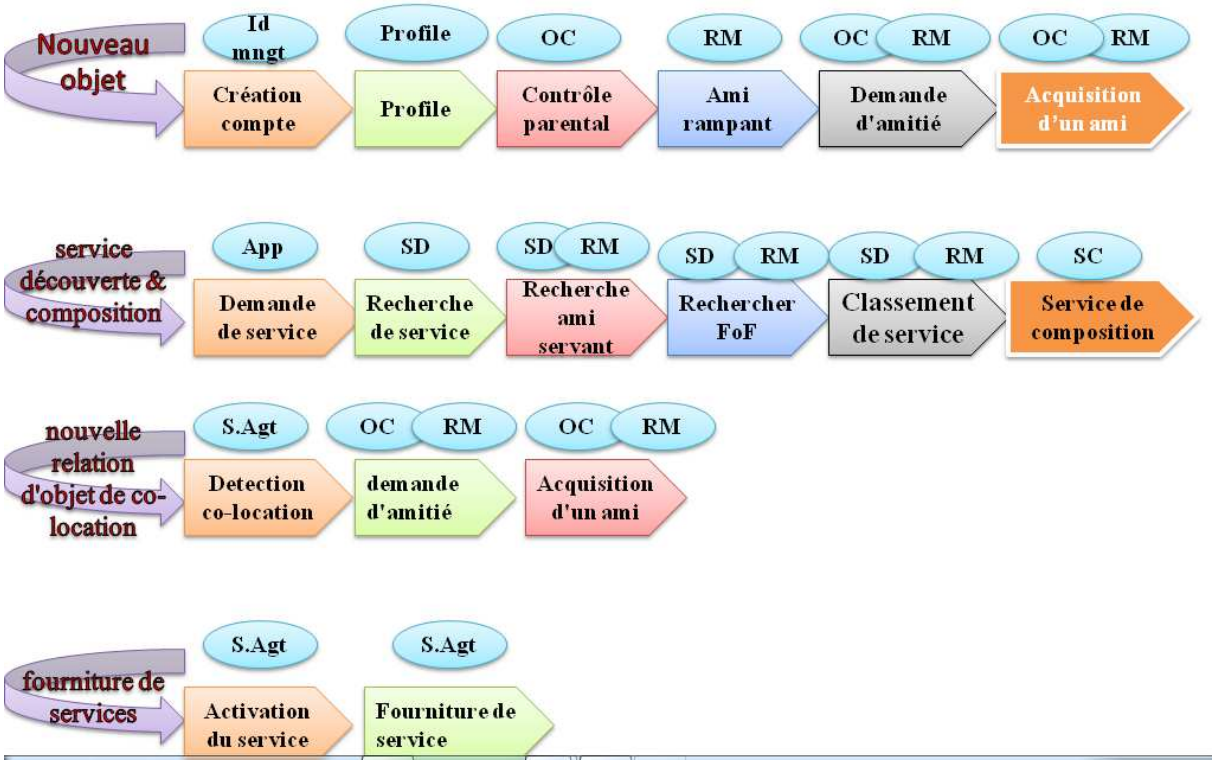

Figure 22: Fonctionnement un nouveau objet dans SIoT

6. Conclusion

L'Internet des objets sociaux (SIoT) modifiera notre façon de communiquer et d'interagir avec le monde. Les applications IoT existantes peuvent être étendues avec une perspective sociale. Les progrès dans la discipline du SIoT sont encore dans un état prématuré. Par conséquent, il existe de nombreuses possibilités de recherche dans ce contexte qui peuvent révolutionner la technologie au cours de la prochaine décennie. SIoT et ses divers aspects y sont désignés tout en délimitant une architecture proposée pour SIoT. Une structure sémantique de SIoT y est expliquée pour améliorer la compréhension des concepts et des services.

Chapitre 4 : Implémentation de SIoT

1. Introduction

Le déploiement d'un réseau exige une étape de simulation avant son installation sur site. La simulation permet de tester à moindre cout les performances d'une solution.

OMNET++ est un environnement de simulation à évènements discrets basé sur le langage C++ : une application open source et sous licence. Il est totalement programmable, paramétrable et modulaire, il a été utilisé avec succès dans divers domaines grâce à son architecture flexible et générique.

OMNET++ sera notre environnement de simulation. Il s'agira d'étendre la simulation grâce à son architecture modulaire en implémentant un nouveau modèle spécifique aux réseaux.

2. Présentation d'OMNet++

OMNET++ est un environnement de simulation à évènements discrets. Utilisé pour la simulation des réseaux de communication et très largement répandu dans divers domaines d'applications tels que :

Figure 23: Le lancement du simulation Omnet++

- La modélisation des protocoles de communications ;
- La modélisation des réseaux filaires et sans fils ;
- La modélisation des systèmes répartis ;
- Les architectures HardWare
- En général, il peut être utilisé pour n'importe quel système à évènements discrets pouvant être modélisé selon des entités communiquant par envoient de messages.

OMNET++ est basé sur la plateforme Eclipse. Il fournit des outils pour la création et la configuration des modèles de réseaux (fichiers NED et INI) et des outils pour l'exécution d'un lot de programmes ainsi que pour l'analyse des résultats de simulation.

3. Description architecturel d'OMNET++

Les modèles OMNET++ constituent en un ensemble de modules hiérarchiquement emboités tel qu'il montré dans la (**figure24).**

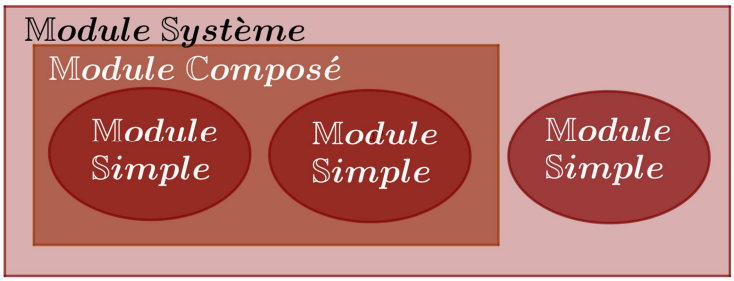

Figure 24: Architecture modulaire de la simulation Omnet++

Les modules simples : sont écrits en C++ en utilisant la librairie de simulation d'OMNET++. Ils contiennent des algorithmes relatifs au modèle implémenté. Leur groupement constitue des modules composés sachant que leurs communications sont gérées grâce à des connexions entre les modules via des « gates (ports) ».

Le module système : créé par l'utilisateur et n'a pas de connexion avec l'environnement extérieur, mais avec ses composants internes (Modules simples et composés).

Les modules peuvent s'attribuer des paramètres assignés aux modules dans les fichiers de description de réseaux (fichier NED) ou encore dans le fichier de configuration « omnetpp.ini ». Ces paramètres sont utiles pour la personnalisation du comportement des modules simples ou encore pour le paramétrage de la topologie des modèles.

4. Installation du simulateur OMNET++

L'installation d'OMNET++ se fait en différentes étapes suivant une procédure d'installation décrite dans le package téléchargé selon le système d'exploitation installé. Les éléments installés sur l'ordinateur seront les suivants :

Etape1 : Cliquez sur le lien suivant pour télécharger OMNeT++ avec MinGW (Il s'agit de la version 5.4.1. Ce package est destiné à Windows et contient le compilateur C++ de MinGW) ; http : //omnetpp.org/omnetpp/download/30-omnet-releases/2307—omnetpp-50-windows

Etape2 : Décompresser le fichier sur l'une des partitions. Soit : C:\ omnetpp-5.4.1 ;

Etape3 : Ouvrir le dossier C:\ omnetpp-5.4.1 et exécuter le fichier mingwenv.cmd ;

Etape4 : Une nouvelle fenêtre s'ouvrira qui ressemble à un shell Linux avec invite de commandes;

Etape5 :

- Taper la commande : ./configure (cette commande vérifiera tous les modules et définira le chemin).
- Taper la deuxième commande : make (cela durera au moins 10 minutes, mais varier d'une machine à l'autre) ;
- Une fois que tout est terminé, vous obtenez un message que tout est réussi et taper omnetpp pour ouvrir l'EDI;
- Veiller installer JDK avant toutes ces étapes, car omnet++I DE est basé sur éclipse qui nécessite l'installation de JDK ;

Etape6 : Pour vérifier la réussite de l'installation, exécuter n'importe quel projet tel que : Tic-Toc;

5. Les principaux fichiers d'OMNET

- **Fichiers (.NED)**

Il Utilise le langage NED de description de réseau et peut être utilisé en 2 modes : mode graphique ou mode Texte qui permet de décrire les paramètres et les ports du module. Les erreurs commises sont indiquées en temps réel par un point rouge situé à gauche du code.

Des exemples de fichier Ned en ses 2 modes sont présentés dans les figures 25 et la26.

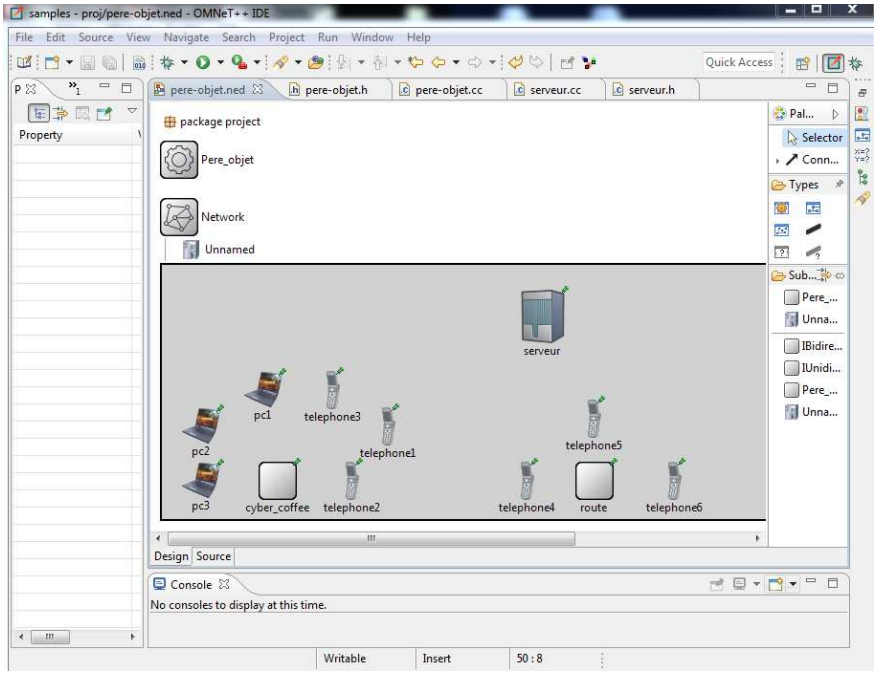

Figure 25: Le fichier NED en mode Graphique

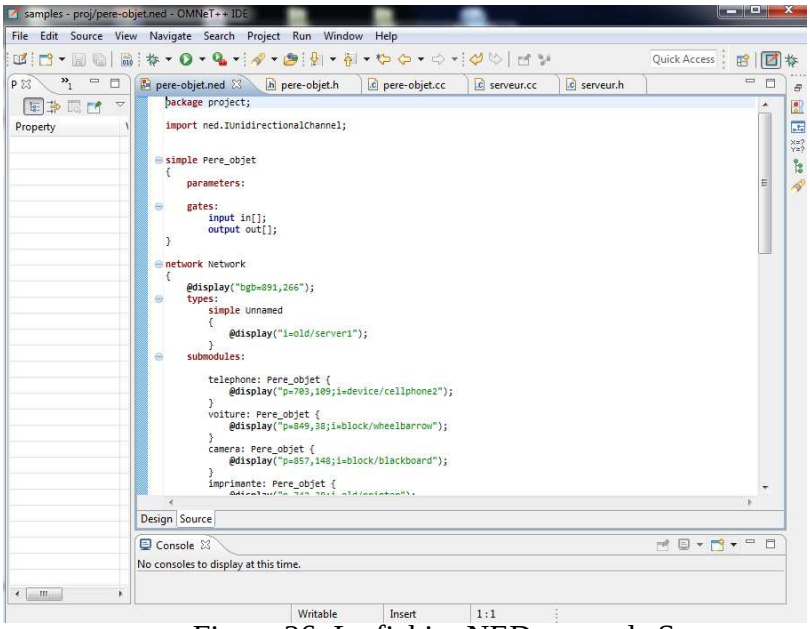

Figure 26: Le fichier NED en mode Source

- **Le fichier (.ini)**

Ce fichier est lié étroitement avec le fichier NED et permet à l'utilisateur d'initialiser les paramètres des différents modules ainsi la topologie du réseau.

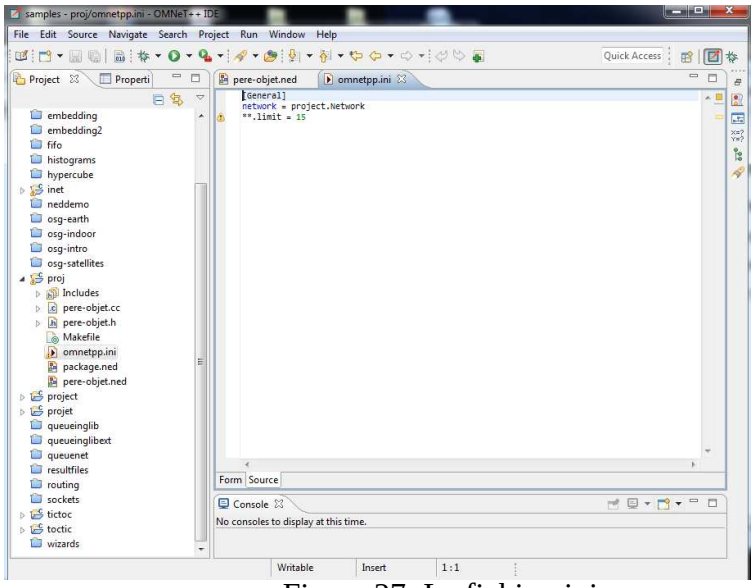

Figure 27: Le fichier .ini

- **Le fichier (.msg)**

Les modules communiquent en échangeant des messages qui peuvent être déclarés dans un fichier d'extension (.msg) où l'on peut ajouter des champs de données. OMNeT++ traduira les définitions de message en C++.

La figure28 est une aide détaillée sur le développement d'exécution d'une simulation sous OMNET.

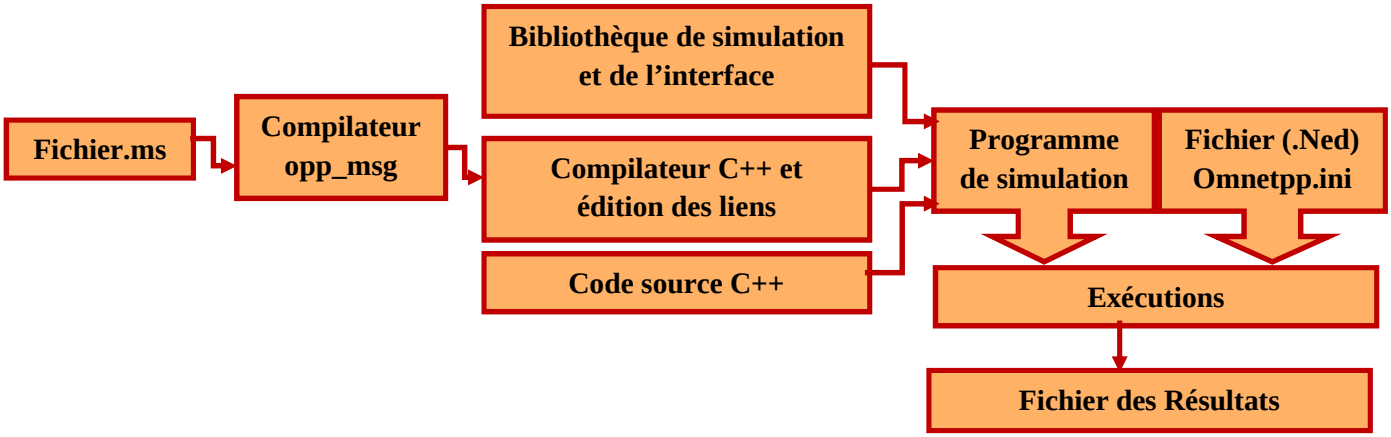

Figure 28: Exécution d'une simulation sous OMNET++

6. Implémentation d'un réseau

- L'OMNeT++ a pris en considération le développement de nouveaux protocoles au sein de cette plateforme. Dans ce contexte, les fichiers templates facilitent la création d'un nouveau réseau avec les différentes couches.
- Pour l'implémentation, il suffit de copier les fichiers nécessaires dans un nouveau répertoire de votre choix.
- Nous avons adopté ce principe et crée un nouveau répertoire appelé (Projet) où nous y avons copié les fichiers : « père-objet.ned », « omnetpp.ini », « package.ned » qui contiennent des racines de procédures, où chacun peut modifier et élaborer son protocole.
- Le programme ne sera complet que par l'ajout et la programmation des fichiers « .cc », « .h » ; « .msg » des différents modules.
- Une fois que tous les fichiers sont complets, la plateforme OMNeT++ prendra en considération ce nouveau réseau en exécutant seulement les étapes de compilation suivantes :
  - ✓ Cliquer droit sur proj puis choisir Build Project dans le menu contextuel qui permet si aucune erreur n'est commise, la création des deux fichiers suivants: Makefile, Include.
  - ✓ Compiler en exécutant le fichier « père-objet .ned ».

Conclusion générale

Le SIoT est l'intégration de 2 technologies : l'IoT et les réseaux sociaux. Il est composé d'objets intelligents connectés socialement, capables de collaborer les uns avec les autres sans aucune intervention humaine. La communication avec et entre eux conduit à la vision d'un paradigme de communication à tout moment, n'importe où, de n'importe quel média ou autre. Cependant, pour remplir la vision du SIoT, il faut rapprocher le concept SIoT de la réalité en mettant au point une architecture rétro-compatible et sécurisée.

Notre étude des littératures a évoqué une vision des interactions entre les objets en visant à mimer les comportements sociaux humains, à les appliquer aux objets communicants et à apporter de nouvelles solutions dans les interactions et les échanges de services entre entités des systèmes intelligents. Ainsi, elle a découvert ces relations et leurs activités, les a identifiées, analysées et classées selon leurs niveaux de confiance. Ces analyses offrent de nouveaux paradigmes d'interaction comme la monétisation de service entre objets, aux problématiques d'économie d'énergie, mais, de nouvelles portes s'ouvrent comme de nouveaux axes de recherche à explorer telles que les questions de sécurité , de confidentialité, de confiance et d'équilibre de service afin de créer réellement une communauté SIoT fiable.

Notre étude a présenté aussi un modèle de l'architecture du concept SIoT basé sur une sorte de relations sociales entre les objets, ainsi; nous découvrîmes ses différentes fonctions telles que : La gestion de relations, la découverte de services, la composition de service, la gestion de la confiance; Nous avons fourni aussi un aperçu des principales architectures proposées dans les différentes littératures, mais nous prévoyons de concevoir une approche appropriée pour faire face aux différents défis de l'IoT à chaque couche du réseau. Nous conclûmes comme d'ailleurs tous les experts qu'il n'existe pas une architecture unique, et que, les concepts se raffineront et les protocoles P2P seront désormais prêts à satisfaire des exigences professionnelles variées.

Nous avons proposé un modèle combinant **l'honnêteté et le crédit** dans le calcul de la confiance. En effet l'utilisation de **l'honnêteté** dans le calcul de la confiance permet non seulement de garantir **une bonne qualité de recommandation** mais aussi la **résistance aux attaques** sur la réputation des nœuds. L'idée du **crédit** permet d'attribuer une valeur aux services et de cette manière encourager l'interaction et éviter les attaques DoS sur les objets.

Les modèles distribués : permettent une autonomie des objets, une très forte évolutivité mais de faible résistance aux attaques, ce qui influence négativement sur

l'exactitude de la dérivation de la confiance et ne garantissent pas le monitorage car il n'existe pas d'entité centrale destinée à la supervision des nœuds du réseau.

Les modèles hiérarchiques : présentent des avantages quant à la possibilité de monitorage, à la faible consommation d'énergie et à la résistance aux attaques, permettant ainsi une grande précision dans le calcul de la confiance.

Aucune solution de celles proposées dans cette analyse ne garantit à la fois l'ensemble des critères préalablement définis et la résistance aux attaques de type DoS, mais, représentent des solutions partielles aux problématiques rencontrées dans la gestion de confiance. Alors, de nombreux défis doivent être relevés avant le déploiement mondial de cette technologie:

1. **Gestion de l'énergie:** Les objets connectés consomment beaucoup d'énergie. Alors, toutes les étapes de la conception des technologies SIoT doivent être orientées vers une consommation d'énergie réduite.
2. **Sécurité, confidentialité et confiance**: Imposer des règles de protection de vie privée, de confidentialité, de sécurité des communications et de fiabilité des interactions.
3. **Opération, gestion et organisation autonomes:** Un fonctionnement automatique est nécessaire pour la plupart des mécanismes de cette vaste plate-forme.
4. **Hétérogénéité**: créer de nouvelles conceptions prenant en charge efficacement des technologies hétérogènes.
5. **Interactions et interfaces:** la manière d'obtenir des données à partir des appareils d'autres utilisateurs n'est pas tout à fait claire à cause des questions de confidentialité.
6. **Développement d'applications**: Toutes les fonctionnalités offertes par SIoT n'ont pas de sens si les applications ne les utilisent pas. L'utilisation d'API sera utile et apportera de nouveaux cas d'utilisation qui contribueront à rendre le SIoT plus fonctionnel.
7. **La gestion des données :** À l'avenir, SIoT créera un océan de data gérable uniquement grâce à l'intelligence artificielle (IA) et notamment le machine « Learning »).
8. **Tolérance aux pannes:** Le bon fonctionnement des différents composants et une organisation architecturale correcte offrent des informations fiables.

Pour que le grand public adopte le SIoT, les prestataires de services devront proposer des applications capables d'améliorer sensiblement la vie des gens. Attendons-nous donc à l'avenir, à la mise sur le marché de tas de nouveaux produits et services, car, avoir le potentiel d'améliorer le monde tel que nous le voudrons : La rapidité à laquelle nous y parviendrons ne dépend que de nous.

Bibliographie

[1] Michael Chorost, "The Networked Pill", MIT Technology Review, 20 mars 2008, http://www.technologyreview.com/biomedicine/20434/?a=f valuable le: 01 Mars 2019

[2] Christopher Trout, Endadget, Source : "Researchers Debut One-Cubic-Millimeter Computer, Want to Stick It in Your Eye ", 26 Ferrier 2011, http://www.engadget.com/2011/02/26/researchers-debut-one-cubic-millimeter-computer-want-to-stick-i/ valuable le: 01 Mars 2019

[3] U. Arijit, B. Soma, J. Joel, B. Vijayanand, and L. Sachin, "Negotiation-based privacy preservation scheme in internet of things platform," Proceedings of the First International Conference on Security of Internet of Things, Kollam, India, 2012.

[4] B. Christophe, M. Boussard, M. Lu, A. Pastor, and V. Toubiana, "The web of things vision: Things as a service and interaction patterns," *Bell Labs Technical Journal (John Wiley & Sons, Inc.)*, vol. 16, no. 1, pp. 55-61, 2011.

[5] B. Fabian, "Secure name services for the Internet of Things," Thèse, Humboldt-Universität zu Berlin, Wirtschaftswissenschaftliche Fakultät, 2008.

[6] A. Nawaf, S. A. Anthony, and A. Akbar Sheikh, "Application of ZigBee and RFID Technologies in Healthcare in Conjunction with the Internet of Things," Proceedings of International Conference on Advances in Mobile Computing ; Multimedia, Vienna, Austria, 2013.

[7] A.-L. Priscila and J.-P. Guillermo, "Collaborative Agents Framework for the Internet of Things," Workshop Proceedings of the 8th International Conference on Intelligent Environments, 2012.

[8] Cisco, "L'Internet des objets (IoT)", 16 déc. 2015; http://www.cisco.com/web/FR/solutions/ trends/IoT/ overview.html. valuable le: 05 Mars 2019

[9] N. Mehdi, "Demain, l'Internet des objets", 18 déc. 2015; http://www.strategie.gouv.fr/publications/demain-linternet-objets. 10 Mars 2019

[10] E. Dave, L'Internet des objets Comment l'évolution actuelle d'Internet transforme-t-elle le monde ?.Cisco IBSG éd.. 2011. 13 p

[11] [H] M. Han and H. Zhang, "Business intelligence architecture based on internet of things " *Journal of Theoretical & Applied Information Technology*, vol. 50, no. 1, pp. 90-95, 2013.

[12] P.-J. Benghozi, S. Bureau, F. Massit-Folléa, C. Waroquiers, and S. Davidson, *L'internet des objets: quels enjeux pour l'Europe*. Éd. de la Maison des sciences de l'homme éd.. 2009. 66 p.

[13] N.Merrani et N.Khimoum. Simulation et évaluation de protocoles de gestion de clés dans les réseaux de capteur. Mémoire d'ingénieur d'état en informatique de l'université Abderrahmane Mira Bejaia, 2009.

[14] Deboleena Dutta, Surajit Das, Chido Tazivazvino, Balakrushna Tripathy, "Social Internet of Things (SIoT): Transforming smart object to social object", Conference Paper · December 2015, http://www.researchgate.net/publication/287216756 (Valable le: 21/04/2019)

[15] Tadrash Shah, Chintan M Bhatt - The Internet of Things: Technologies, Communications and Computing, CSI Communications, VOL No. 38 April 2014

[16] S. Atmani. Protocole de sécurité Pour les Réseaux de capteurs Sans Fil. PhD thesis, Université de Batna 2. Juillet 2010.

[17] C.Llorens, L.Levier, D.Valois et B.Morin. Tableaux de bord de la sécurité réseau. Editions Eyrolles, 2011.

[18] MR.Abdmeziem, D.Tandjaoui et I.Romdhani. A decentralized batch-based group key management protocol for mobile internet of things (dbgk). In Computer and Information Technology ; Ubiquitous Computing and Communications ; Dependable, Autonomic and Secure Computing ; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), 2015 IEEE International Conference on, pages

[19] R. Chen, F. Bao, and J. Guo. Trust-based service management for social internet of things systems. IEEE Transactions on Dependable and Secure Computing, PP(99) :1-1, 2015.

[20] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions." Future Generation Computer Systems, vol. 29, no. 7, pp. 1645-1660.

[21] J. D. Pessemier, "Une réflexion sur « L'internet des Objets » (IdO) ou « Internet of Things » (IoT)," 2015.

[22] les experts Ooreka, "Système RFID : définition et fonctionnement d'un système RFID ", http://rfid.comprendrechoisir.com/comprendre/systeme-rfid valable 22 01. 2019:.

[23] J. A. Stankovic, "Wireless sensor networks," IEEE Computer Society, vol. 41, no. 10, pp. 92-95, 2008.

[24] N. Daniel, R. Marcel, and K. Daniel, Livre blanc Machine To Machine enjeux et perspectives: Orange Business Services, Syntec informatique, Fing, 2006. 40 p.

[25] D. Guinard and V. Trifa, "Towards the web of things: Web mashups for embedded devices," Workshop on Mashups, Enterprise Mashups and Lightweight Composition on the Web (MEM 2009), in proceedings of WWW .International World Wide Web Conferences. Madrid, Spain, 2009.

[26] P. Spiess, S. Karnouskos, D. Guinard, D. Savio, O. Baecker, L. Souza, *et al.*, "SOA-based integration of the internet of things in enterprise services." Web Services, 2009. ICWS 2009. IEEE International

[27] V. H. Vu. Infrastructure de gestion de la confiance sur internet. PhD thesis, École Nationale Supérieure des Mines de Saint-Etienne, 2010.

[28] A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on, pages 9-pp. IEEE, 2000.

[29] M. Nitti, R. Girau, and L. Atzori. Trustworthiness management in the social internet of things. IEEE Transactions on Knowledge and Data Engineering, 26(5) :1253-1266, May 2014.

[30] E.Bajic and O. Hajlaoui, Apports des paradigms sociaux dans l'internet des objets industriels: vers des objets communicants industriels sociaux, 12e Conférence Internationale de Modélisation, Optimisation et Simulation, 27 au 29 juin 2018 – Toulouse, France

[31] R. Neisse, M. Wegdam, and M. van Sinderen. Trust management support for context-aware service platforms. In User-Centric Networking, pages 75-106. Springer, 2014.

[32] Recommendation X ITU-T. 509 âoe. Draft Revised ITU-T Recommendation X, 509 :9594-8,1997.

[33] J. Guo and I. R. Chen. A classification of trust computation models for service-oriented internet of things systems. In SCC, pages 324-331. IEEE Computer Society, 2015.

[34] A. Jsang, C. Keser, and T. Dimitrakos. Can we manage trust ? In Trust management, pages 93-107. Springer, 2005.

[35] T. Schlossnagle. Scalable internet architectures. Pearson Education, Londres, 2006.

[36] A. Aris, S. F. Oktug, and S. B. O. Yalcin. Internet-of-things security : Denial of service attacks. In Signal Processing and Communications Applications Conference (SIU), 2015 23th, pages 903-906. IEEE, 2015.

[37] https://www.w3.org/2014/02/wot/papers/baccelli.pdf (accédé le 02/03/2019).

[38] S. Abbas, M. Merabti, and D. Llewellyn-Jones. Deterring whitewashing attacks in reputation based schemes for mobile ad hoc networks. In Wireless Days, pages 1-6. IEEE, 2010.

[39] A. Beghriche and A. Bilami. Modélisation et gestion de la confiance dans les réseaux mobiles ad hoc. In CIIA, 2009.

[40] Eddy Bajic , Oussama Hajlaoui , Apports des paradigmes sociaux dans l'internet des objets industriels : vers des objets communicants industriels sociaux . Conference Paper - June 2018

[41] Wu, J., Dong, M., Ota, K., et al. Securing distributed storage for social internet of things Using regenerating code and blom key agreement. Peer-to-Peer Netw Appl **8**(6), 1133–1142, (2015)

[42] G. Zacharia and P. Maes. Trust management through reputation mechanisms. Applied Artificial Intelligence, 14(9) :881-907, 2000.

[43] N. BUDAN, B. TEDESCHI And S. VAUBOURG. Nouvelles Technologies Réseau, Les réseaux peer-to-peer, Fonctionnement, exemples, limites-2003 http://igm.univ-mlv.fr/~duris/NTREZO/20022003/Peer-to-peer.pdf (Available online: 03/03/2019)

[44] G. Lize, W. Jingpei, and S. Bin. Trust management mechanism for internet of things. China Communications, 11(2) :148-156, Feb 2014.

[45] H. Syphax. Vers un modèle de confiance pour l'Internet des Objets. En vue d'obtention du diplôme de Master Recherche en Informatique, Option : Réseaux et Systèmes Distribués. Université Abderrahmane MIRA , Béjaia. PP.29-36,2016

[46] Y. Ben Saied, A. Olivereau, D. Zeghlache, and M. Laurent. Trust management system design for the internet of things : A context-aware and multi-service approach. Computers & Security, 39, Part B :351 - 365, 2013.

[47] Tripathy, B.K., Dutta D., Tazivazvino, Chido.: (Ed.) Internet of Things (IoT) in 5G Mobile Technologies: On the Research and Development of Social Internet of Things. 153–173, Springer International

[48] H. Xiao, N. Sidhu, and B. Christianson. Guarantor and reputation based trust model for social internet of things. In Wireless Communications and Mobile Computing Conference (IWCMC), 2015 International, pages 600{605, Aug 2015.

[49] F. Bao, I. R. Chen, and J. Guo. Scalable, adaptive and survivable trust management for community of interest based internet of things systems. In Autonomous Decentralized Systems (ISADS), 2013 IEEE Eleventh International Symposium on, pages 1-7, March 2013.

[50] P. N. Mahalle, P. A. Thakre, N. R. Prasad, and R. Prasad. A fuzzy approach to trust based access control in internet of things. In Wireless Communications, Vehicular Techno-logy, Information Theory and Aerospace Electronic Systems (VITAE), 2013 3rd International Conference on, pages 1-5, June 2013.

[51] I. R. Chen, F. Bao, and J. Guo. Trust-based service management for social internet of things systems. IEEE Transactions on Dependable and Secure Computing, PP(99) :1-1, 2015.

[59] M. Varguez-Moo, F. Moo-Mena, and V. Uc-Cetina, "Use of classification algorithms for semantic web services discovery," Journal of Computers, vol. 8, no. 7, pp. 1810-1814, 2013.

[60] F. Paganelli and D. Parlanti, "A DHT-Based Discovery Service for the Internet of Things," Journal of Computer Networks & Communications, pp. 1-11, 2012.

[61] V. Altmann, J. Skodzik, P. Danielis, J. Mueller, F. Golatowski, and D. Timmermann, "A DHT-Based Scalable Approach for Device and Service Discovery," Embedded and Ubiquitous Computing (EUC), 2014 12th IEEE International Conference on, 2014.

[62] M. Makhlughian, S. M. Hashemi, Y. Rastegari, and E. Pejman, "Web service selection based on ranking of qos using associative classification," International Journal on Web Service Computing (IJWSC), vol. 2, pp. 1-14, 2012.

[63] S. Wang, Z. Zheng, Q. Sun, H. Zou, and F. Yang, "Cloud model for service selection," Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on, 2011.

[64] R. Mohana and D. Dahiya, "Approach and impact of a protocol for selection of service in web service platform," *SIGSOFT Softw. Eng. Notes*, vol. 37, no. 1, pp. 1-6, 2012.

[65] B. Jeong, D. Lee, J. Lee, and H. Cho, "Support for seamless data exchanges between web services through information mapping analysis using kernel methods," *Expert Systems with Applications*, vol. 36, no. 1, pp. 358-365, 2009.

[66] J. Xu, G. Xiao, J. Lu, Q. Liang, and J. Shen, "Customizable data exchange based on Web service," e Business Engineering, 2009. ICEBE'09. IEEE International Conference on, 2009.

[67] S. Kubler, I. D. Nargund, K. Fr, #228, mling, and W. Derigent, "Peer-to-Peer Data Synchronization Agents," Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT) - Volume 03, 2014.

[68] Chiang, C.C.; Wu, H.K.; Liu, W.; Gerla, M. Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel. In Proceedings of the IEEE SICON, Singapore, 14–17 April 1997; pp. 197–211.

[69] Nitti, M.; Atzori, L.; Cvijikj, I.P. Friendship Selection in the Social Internet of Things: Challenges and Possible Strategies. IEEE Internet Things J. **2015**. 2. 240–247.

[70] Nitti, M.; Pilloni, V.; Colistra, G.; Atzori, L. The Virtual Object as a Major Element of the Internet of Things: A Survey. IEEE Commun. Surv. Tutor. **2016**, 18, 1228–1240.

[71] Ji Eun Kim, Adriano Maron, and Daniel Mosse. 2015. Socialite: A Flexible Framework for Social Internet of Œings. In Mobile Data Management (MDM), 2015 16th IEEE International Conference on, Vol. 1. IEEE, 94–103.

[72] Chen R., Guo J., Bao F.: Trust management for soa-based IoT and its application to service composition. IEEE Trans. Serv. Comput. 2(1), 1 (2015)

[73] Byun, J., Kim, S. H., Kim, D.,: Lilliput: Ontology-Based Platform for IoT Social Networks. In: IEEE International Conference on Services Computing, Anchorage, AK, 139–146 (2014)

[74] Kang, D. H., Choi H. S., Rhee, W. S.: Social Correlation Group generation mechanism in social IoT environment. In: Eighth International Conference on Ubiquitous and Future Networks (ICUFN), Vienna, 514–519 (2016)

[75] Luigi Atzori,Antonio Iera, GiacomoMorabito: SIoT: Giving a Social Structure to the Internet of Things, IEEE Communication Letters. VOL.15. NO.11. November 2011

[76] A. M. Ortiz, Dina Hussein, Soochang Park, Son N. Han, Noel Crespi: The Cluster Between Internet of Things and SocialNetworks: Review and Research Challenges, IEEE Internet of things journal, VOL. 1, NO. 3, June 2014

[77] Atzori, L., Iera, A., Morabito, G., Nitti, M.: The social internet of things (SIoT)–when social networks meet the internet of things: concept, architecture and network characterization. Comput. Netw. **56**(16), 3594–3608 (2012)

[78] Kleinberg, J.: The small-world phenomenon: an algorithmic perspective. In: Proc. ACM Symposium on Theory and Computing (2000)

[79] Fiske, A. P.: The four elementary forms of sociality: framework for a unified theory of social relations. Psychological review **99**, 689–723 (1992)

[80] Nitti, M., Atzori, L., Cvijikj, I. P.: Friendship Selection in the Social Internet of Things: Challenges and Possible Strategies. IEEE Internet of Things Journal **2**(3), 240–247 (2015)

[81] Boguna, M., Krioukov, D., Claffy, K.C.: Navigability of complex networks. Nat. Phys. **5**(1), 74–80

[82] Abdelghani, W., Zayani, C.A., Amous, I., Sèdes, F.: Social Media: The Good, the Bad, and the Ugly: Trust Management in Social Internet of Things: A Survey. Lecture Notes in Computer Science, Springer International Publishing, 430–441 (2016)

[83] Zheng L., et al. (Ed.): Technologies, applications and governance in the Internet of things - Global Technological and Societal Trends. River Publisher (2011)

[84] http://protege.stanford.edu/