3-30-2021

A Plug&Play approach for modeling and simulating applications in the era of internet of social things

Zakaria Maamar
Zayed University

Mohamed Sellami
Institut Mines-Télécom

Fatma Masmoudi
Prince Sattam Bin Abdulaziz University

Muhammad Asim

Abdul Haseeb
Liverpool John Moores University

See next page for additional authors

Recommended Citation

Maamar, Zakaria; Sellami, Mohamed; Masmoudi, Fatma; Asim, Muhammad; Haseeb, Abdul; Baker, Thar; and Yahya, Fadwa, "A Plug&Play approach for modeling and simulating applications in the era of internet of social things" (2021). *All Works*. 4170.
https://zuscholars.zu.ac.ae/works/4170

Author First name, Last name, Institution

Zakaria Maamar, Mohamed Sellami, Fatma Masmoudi, Muhammad Asim, Abdul Haseeb, Thar Baker, and Fadwa Yahya

Received: 27 August 2020 | Revised: 29 October 2020 | Accepted: 2 November 2020

IET Smart Cities

DOI: 10.1049/smc2.12005

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH PAPER

A Plug&Play approach for modeling and simulating applications in the era of internet of social things

Zakaria Maamar[1] | **Mohamed Sellami**[2] | **Fatma Masmoudi**[3,4] | **Muhammad Asim**[5] | **Abdul Haseeb**[6] | **Thar Baker**[6] | **Fadwa Yahya**[3]

[1]Zayed University, Dubai, UAE

[2]Samovar, Télécom SudParis, Institut Polytechnique de Paris, Palaiseau, France

[3]Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al Kharj, Saudi Arabia

[4]ReDCAD Laboratory, University of Sfax, Tunisia

[5]FAST-NUCES, Islamabad, Pakistan

[6]Liverpool John Moores University, Liverpool, UK

Correspondence

Mohamed Sellami, Samovar, Télécom SudParis, Institut Polytechnique de Paris, France.
Email: mohamed.sellami@telecom-sudparis.eu

Abstract

This article presents an approach to model and simulate Plug&Play social things. Confined into silos, existing (not social) things are restricted to basic operations like sensing and actuating, which deprive them from participating in the satisfaction of complex business applications. Contrarily, social things are expected to engage in collaborative scenarios and to tap into specific relations that connect them to peers when achieving these scenarios. These relations are referred to as complimentary, antagonism, and competition, and allow to develop networks of things. To capitalize on such networks, the approach to model and simulate Plug&Play social things puts forward four stages that are referred to as connecting to demystify social relations between things, influencing to examine the impact of social relations on things, playing to make things perform while considering influence, and incentivizing to reward things based on their performance. A smart system for elderly the care centers has been developed to showcase the technical doability of Plug&Play social things. The system is an integrated development environment allowing IoT engineers to define the collaboration of social things, thanks to a set of drag&drop operations.

1 | INTRODUCTION

Internet-of-Things (IoT), a disruptive technology that combines many Information and Communication Technologies (ICT), has become an integral part of people's daily lives [1,2]. According to Gartner[1], 6.4 billion connected things were in use in 2016, up by 3% from 2015, and will reach between $3.9 trillion and $11.1 trillion per year by 2025[2]. Like other ICT (e.g. business process management systems and Web services), IoT is also surfing the *social wave* (with reference to Web 2.0) to tap into the opportunities of using Web 2.0 technologies and applications such as capturing and drilling into the massive amount of data that users generate over social media (e.g. Facebook and WhatsApp). An objective of this drill is to make software systems mimic users, which should lead to better personalized systems. In the ICT literature, surfing the social wave has resulted into a plethora of technical terms (sometimes ‘buzzwords’) such as, Internet of Social Things (IoST) [3,4], social Internet of Things [5], social Web services [6,7], social business processes [8,9], social Web of things [10], and social cloud [11].

In line with the social trend (sometimes ‘fever’), we also witness in the ICT community some positive signs of empowering things with extra capabilities, thanks to initiatives related to intelligent things [12], wisdom Web of things [13], semantic things [14], agents of things [15,16], agentification of IoT [17], process of things [18], and organizational structures for IoT [19]. In this article, we argue that a successful thing empowerment should be aligned to a set of stages that would bring separate things together. Confined into silos and subject to many computation, communication, and storage restrictions, things can no longer afford this confinement nor continue to ‘struggle’ with these restrictions. Things' collective behaviors are deemed necessary to address users' changing and complex requirements. So, how to allow things to work together? To respond to this question, we suggest a four-stage approach that would revolve around the concepts of *connecting*, *influencing*, *playing*, and

[1] *tinyurl.com/y4pkvghv*.

[2] shorturl.at/jGZ09

incentivizing. Connecting demystifies social relations between things. Influencing examines the impact of social relations on things. Playing makes things perform while considering this impact. Finally, incentivizing rewards things based on their performance. These four stages should promote a new generation of things that we would refer to as Plug&Play. Our objective is to allow things to know with whom to collaborate, whom to avoid, how to react, and what to request when they participate in collaborative scenarios.

On top of demystifying the potential social relations between things, another objective is to assign the usage properties such as, limited and occurrence, to these relations. This should allow to define when and where social relations are activated, for how long, and for how many times. These properties would permit to avoid resource starvation in an environment where millions of things would operate and hence, compete for resources. They are drawn from our previous work on social coordination of business processes [20] and consumption properties of (cloud) resources [21] and would regulate the lifecycles of social relations in terms of longevity (short-term versus long-term), nature (static versus dynamic), and occurrence (one versus multiple). Our contributions are as follows:

1. Definition of Plug&Play things in the context of IoT. Contrarily to existing practices that confine things into silos, Plug&Play things would be connected together through specific social relations that would support their participation in collaborative sessions.
2. Analysis of the impact of social relations on things' behaviors. Using these relations, things would 'know' with whom they would end-up collaborating, co-existing, and competing.
3. Control over social relations using a set of usage properties referred to as un/limited, limited-but-renewable/expandable, and single/multiple occurrence.
4. And, development of a system to demonstrate Plug&Play things in the context of smart care centers for elderly people.

The rest of this article is organized as follows: 1. Section 2 discusses blending social computing with IoT and presents a case study. 2. Section 3 defines the core concepts of Plug&Play things namely social relations, usage properties, and influence. The four stages of connecting, influencing, playing, and incentivizing are also detailed in this section. Finally, 3. Section 4 concludes the paper and presents future work.

2 | BACKGROUND

This section discusses some initiatives about blending social computing with IoT and then, presents a case study that would motivate the adoption of Plug&Play things.

2.1 | Social computing and IoT

Like many emerging ICT topics, there is not a consensus on what is IoT. According to Klein et al., IoT *'refers to an emerging paradigm consisting of a continuum of uniquely addressable things communicating to one another to form a worldwide dynamic network'*. Others consider IoT as an Internet-based information architecture that provides technical means to interconnect things and enrich them with capabilities [22–24]. Capabilities include communication and cooperation, addressability, identification, sensing, actuation, embedded information processing, localization, user interfaces, monitoring, control, optimization, and autonomy.

According to Pticek et al., embracing the social wave would mean connecting everyone and everything [25]. A good review of Social IoT (not to confuse with IoST) is presented in [26]. Atzori et al. consider things as intelligent objects and suggest that models are used for studying social networks of humans can be extended to social networks of objects [3,27]. Networks of objects could be built upon relations such as parental (similar objects built in the same period by the same manufacturer), co-location (objects in the same venue), co-work (objects participating in the same scenario), ownership (objects having the same user), and social (when objects come into contact sporadically or continuously). Atzori et al. also discuss the paradigm shift which happens from human-object interaction to object-object interaction.

In another work, Ortiz et al. shed light on the challenges and issues that undermine the blend of social computing with IoT [4]. These challenges and issues include defining a social-thing architecture, addressing interoperability of things, considering new business models, discovering things, managing energy consumption of things, handling security, privacy, and trust of things. A social-thing architecture would consist of actors (smart things and humans), an intelligent system to manage actors' interactions, an interface for actors to engage in interaction, and the Internet as a means to support interaction. In support to Ortiz et al.'s work [4], Hussain et al. examine cyber-physical systems from a social perspective and note that existing approaches do not recognize how such systems can be socially connected so, that, they interact in collaborative decision-making like humans [28]. The authors suggest a software agent-centric Semantic Social-Collaborative Network (SSCN) that provides functionality to represent and manage cyber-physical resources in a social network. SSCN is associated with an extended ontology model to semantically describe human and non-human resources and their social interactions, and has been demonstrated in the context of UK mental-health services.

Although the blend of social computing with IoT is still in 'gestation', another trend known as IoST is rising and needs to be contrasted to Social-Internet-of-Things [29]. On the one hand, things in the Social Internet need to be configured before they join networks that would be developed upon social relations like those suggested by Atzori et al. [27]. Here, the social Internet offers the necessary protocols to set-up, manage, and maintain the networks of things. On the other hand, social things would need capabilities to 'crawl' the networks of things when looking for partners, assessing partners' capabilities, and forming alliances with partners.

The aforementioned paragraphs provided an overview of some works blending social computing with IoT. This blend requires specific mechanisms that would ensure identifying the relevant things according to situations' requirements. We build these mechanisms upon the principle of Plug&Play where things will be connected together and then, put into action. By analogy with people, things can collaborate when offering complimentary functionalities and can compete when offering similar functionalities. On top of connecting everyone, the trend also is to connect everything.

2.2 | Case study

Our case study discusses the benefits of Plug&Play social things to elderly people who require permanent assistance and monitoring. Many studies confirm that population ageing is a dominant global demographic trend of the 21st century[3]. We could think of many case studies where both networks of persons and networks of things would co-exist provisioning value-added services to elderly people. Networks of persons like Facebook and WhatsApp are well established counting on persons' judgments on when to sign in, sign out, post, comment, to cite just some. Contrarily, networks of things barely exist due to other pressing concerns that are slowing down the adoption of IoT. These concerns are diversity and multiplicity of things' development and communication technologies, users' reluctance to and sometimes rejection of things because of privacy invasion, limited IoT-platform interoperability, and passive nature of things that primarily act as data suppliers (with some actuating capabilities). Networks of things would have a role in exemplifying Plug&Play social things. It is worth mentioning that networks of software components like Web services are already reported in the literature [7,30].

In a smart center for elderly people, case studies that would demonstrate the support of Plug&Play social things to these people could be as follows. First, a remote control could form a temporary relation with the TV according to an elderly person's watching habits when she is in the living room and between 5:00 PM and 9:00 PM. Temporary, since the same remote control could be used for other purposes like opening and closing the blinds in a particular room, which means forming another temporary relation between the remote and blinds when the TV is off. In the same context, the remote-control use would be restricted to the living room where the TV is installed. Beyond the living room, the relation between the TV and remote is disabled. Another case of Plug&Play social things is a smart watch that could instruct an automatic medicine dispenser to prepare medicine doses according to an elderly person's blood-pressure level. The relation between the watch and dispenser would be set according to the duration of the treatment but could be extended, should the treatment be renewed.

The two aforementioned cases highlight potential social relations between things (TV, remote control, smart watch, and medicine dispenser) along with some properties that could characterize these relations like, between whom and whom, are these relations permanent or temporary, and where and/or when do these relations become activated?

3 | PLUG&PLAY SOCIAL THINGS

This section first, sheds light on potential social relations between things. Then, it discusses how we assign usage properties to these social relations to ensure their proper use. Finally, the section examines how things influence each other when they engage in collaborative scenarios. Implementation details of these scenarios are also discussed in this section.

3.1 | What social relations between things?

To expose social relations in the context of Plug&Play social things, we rely on Section 2.1's discussions and our own relations defined in previous projects on thing agentification [31] and social Web services [30]. Our proposed social relations allow connecting things (t) from three perspectives: recommendation that we associate with *complimentary* relation, opposition that we associate with *antagonism* relation, and exclusion that we associate with *competition* relation.

1. Complimentary(t_i, t_j) is about the "joint" participation of things, for example, TV and remote control, in satisfying users' demands (ud). Equation (1) assesses the complimentary level between t_i and t_j where acceptedRec$_{ud}$(t_i, t_j) is the number of times that t_i's recommendations for t_j are accepted by the IoT engineer and madeRec$_{ud}$(t_i, t_j) is the number of times that t_i recommended t_j (including the declined recommendations, declinedRec$_{ud}$(t_i, t_j)). A high complimentary level indicates a positive synergy between things.

$$w_{complementary(t_i,t_j)} = \frac{acceptedRec_{ud}(t_i,t_j)}{madeRec_{ud}(t_i,t_j)} \tag{1}$$

2. Antagonism(t_i, t_j) is about the 'sensitivity' that arises between things, for example, a coffee maker and espresso machine, when they jointly participate in satisfying the users' demands. Equation (2) assesses the antagonism level between t_i and t_j where joint$_{ud}$(t_i, t_j) is the number of times that t_i and t_j jointly participated in satisfying users' demands and participated$_{ud}$(t_i|¬t_j) is the number of times that t_i participated in satisfying users' demands without t_j and vice versa. A high antagonism level indicates a strong coupling between things.

$$w_{antagonism(t_i,t_j)} = \frac{joint_{ud}(t_i,t_j)}{participated_{ud}(t_i|\neg t_j) + participated_{ud}(t_j|\neg t_i)} \tag{2}$$

3. Competition(t_i, t_j) is about the 'exclusion' between things, for example, either cordless phone or regular phone, as one thing, only, can participate in satisfying a user's demand.

[3] www.weforum.org/agenda/2019/10/ageing-economics-population-health.

Equation (3) assesses the competition level between t$_i$ and t$_j$ where selected$_{ud}$(t$_i$, t$_j$) is the number of times that t$_i$ is selected over t$_j$ to participate in satisfying users' demands and possible$_{ud}$(t$_i$, t$_j$) is the number of times that both t$_i$ and t$_j$ are potential candidates for participation in satisfying users' demands. A high competition level indicates the relevancy of selecting a thing over another.

$$w_{competition(t_i,t_j)} = \frac{selected_{ud}(t_i, t_j)}{possible_{ud}(t_i, t_j)} \tag{3}$$

We illustrate the three afore-mentioned social relations in Section 3.4. For instance, a thing could provide extra services, thanks to the network of complementary things. Also, a thing could improve its performance thanks to the network of competing things.

3.2 | What usage properties for social relations?

To have a 'controlled' use of social relations between things in terms of why, where, and when these relations become activated, we assign them usage properties. We build upon our previous work on social coordination of business processes [20] and consumption properties of (cloud) resources [21] to suggest six usage properties (some are exclusive): unlimited (ul), limited (li), limited-but-renewable (lr), limited-but-expandable (le), single-occurrence (so), and multiple-occurrence (mo). Unless stated, a social relation is by default ul (exists forever between a couple of things) and/or mo (concurrently exists between many couples of things).

Figure 1 is a state diagram of an ul social relation's usage cycle. More/Some states will be added/dropped to/from this diagram depending on the characteristics of other usage properties. On the one hand, the states (s_i) include not-formed (the relation is not created, yet), formed (the relation is now created so the things become connected), used (the relation is being exploited by the things participating in the relation), and dismantled (the relation stops existing after unbinding all the things that were participating in the relation). On the other hand, the transitions (trans$_i$) connecting the states include start, waiting-to-be-used, and pulling-out.

Li means that forming a social relation between two things and is restricted to a time period and/or location. Beyond this time period and/or location, the relation is automatically dismantled. Contrarily, ul allows a social relation to remain activated until one or both things decide to pull out from the relation leading to dismantling this relation. Building upon Figure 1, Figure 2 is a state diagram for the usage cycle of li social relation. Highlights in this diagram are two transitions, unused(conditions-unmet) that puts an end to the social relation's existence without being used and use-completion (conditions-unmet) that also puts an end to the social relation's existence but this time after being used. In either case, the limitedness's conditions become unmet due to for instance, expiry date and non-agreed upon location.

Lr means that forming a social relation between two things. It remains activated beyond the (initial) agreed-upon time period and, that, both things agree on extending this time period as they see fit, until one or both things decide to pull out from this relation leading to its dismantlement. Le means that forming a social relation between two things remains activated beyond the (initial) agreed-upon location and, that, both things agree on adding more locations as they see fit, until one or both things decide to pull out from this relation leading to its dismantlement. Building upon Figure 2, Figure 3 is a state diagram for the usage cycle of lr/le social relation. Highlights

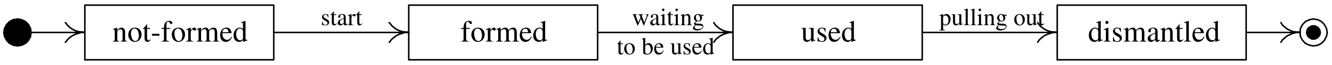


FIGURE 1 Unlimited social relation's usage cycle as a state diagram

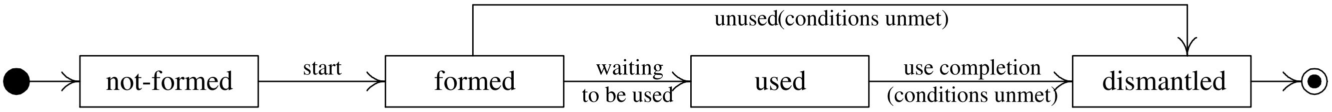


FIGURE 2 Limited social relation's usage cycle as a state diagram

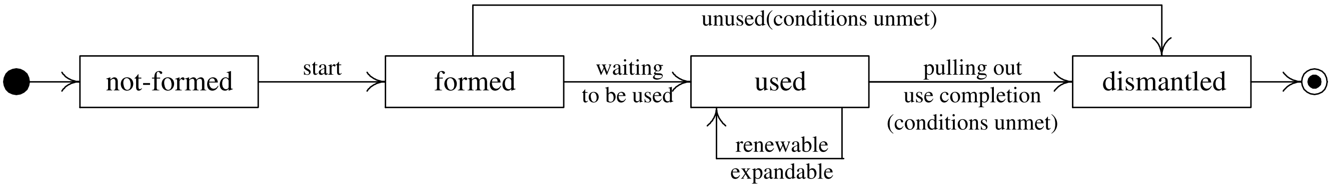


FIGURE 3 Limited-but-renewable/expandable social relation's usage cycle as a state diagram

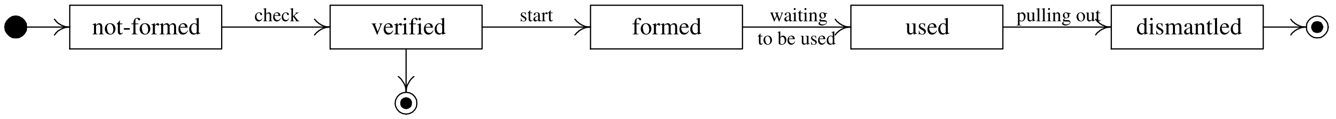


FIGURE 4 Single-occurrence social relation's usage cycle as a state diagram

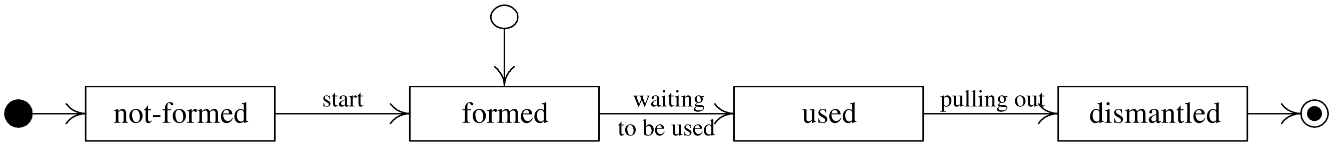


FIGURE 5 Multiple-occurrence social relation's usage cycle as a state diagram

in this diagram are two transitions, pulling-out that puts an end to the existence of the social relation following the decision of participating things to unbind from this relation and renewable/expandable that allows the social relation to continue existing following the decision of participating things to extend this relation.

So means that a social relation in an ecosystem of things is present once (i.e. uniqueness). Building upon Figure 1, Figure 4 is a state diagram for the usage cycle of so social relation. Highlights in this diagram are one state, verified that ensures the uniqueness of the social relation and one transition, check that initiates the counting of the occurrences of the social relation.

Mo allows a social relation to be present many times in an ecosystem of things. Building upon both Figures 1 and 4, Figure 5 is a state diagram for the usage cycle of a mo social relation. Highlights in this diagram are one entry state that allows the formation of the social relation multiple times.

We formally define a social relation and its usage property-based usage cycle as follows.

Definition 3.1 A social relation sr is defined by the tuple < id, name, P, up, UC_{up}> where id is the identifier of the social relation, name is the name of the social relation, P is the set of participating things in the social relation with $|P| = 2$, up is the usage property assigned to the social relation where up $\in$\{li, ul, lr, le, so, mo\}, and $UC_{up} = \{uc_{up}\}$ is the set of all usage cycles (Definition 3.2) associated with the usage property of the social relation.

Definition 3.2 A social relation's usage cycle sr_{uc} for a usage property up is defined by the couple <S, T> representing the state diagram of the social relation with respect to this usage property, that is, $sr_{uc_{up}} = s_i$ $trans_i \rightarrow s_{i+1}\ trans_{i+1} \rightarrow s_{i+2} \ldots s_{j-1}\ trans_{j-1} \rightarrow s_j$ where S is the set of states $\{s_i\}$ and T is the set of transitions $\{trans_i\}$ connecting the states together.

To track the progress of a usage cycle at run-time (r), we define S_{active} that is the set of all states that have been taken on since the activation of the usage cycle; $S_{\text{active}} \subset S$ and $S_{\text{active}} = \{\text{not} - \text{formed}\}$ at initialization time. For illustration the usage cycle for an ul social relation is $sr_{uc_{ul}} = \text{not} - \text{formed}$ start $\rightarrow$ formed waiting−to−be−used $\rightarrow$ used pulling−out $\rightarrow$ dismantled.

3.3 | How do things influence each other?

In addition to complimentary, antagonism, and competition social-relations, we examine how things could influence each other when they are connected through these relations. We refine influence into positive, negative, and neutral. Positive and negative influences have a direct impact on things when it comes to the goals they pursue, the beliefs they manage, the tasks they perform, and the resources they consume.

In [32], we examined influence in the context of multi-agent systems and developed the Goal-Belief-Task-Resource (GBTR) framework[4]. This framework is in line with Castelfranchi and Conte's statement that agent interaction is not about exchanging information, only, but an instrument that influences others since they change their goals and adopt one's goals [33]. Table 1 summarizes the way we foresee the application of the GBTR framework to Plug&Play social things (t_i and t_j). We decompose influence relations into two categories. The first category happens between goals of things and triggers the formation of (*i*) additional relations between things and tasks and (*ii*) additional relations between things and resources. The second category happens between beliefs of things. The way we use influence relations in conjunction with social relations is presented in Section 3.4.

In Table 1, 'outcome' column includes eight relations that differentiate positive from negative influence on things. These relations are *facilitate/hinder* between goals, *affirm/contradict* between beliefs, *perform/work-for* between things and tasks, and *offer/take-over* between things and resources. Similar relations are adopted in multi-agent systems with the work of Decker and Lesser on coordinating partial-global plans [34]. The objectives of this coordination are to avoid redundant activities, shift activities to idle executors, and provide predictive results.

[4] *G for what to pursue, B for what to know, T for what to do, and R for what to consume.*

TABLE 1 Types of influence between things (adopted from [32])

Influence	Type	Description	Outcome
Goal	Positive (+)	t_i has goals that support t_j achieve its own goals	Facilitate relation between things' goals
	Negative (−)	t_i has goals that delay t_j from achieving its own goals	Hinder relation between things' goals
Resource	Positive (+)	t_i offers some of its resources to t_j helping it achieve its goals	Offer relation between things and resources
	Negative (−)	t_i takes over some of t_j's resources making it lack these resources to achieve its goals	Take-over relation between things and resources
Task	Positive (+)	t_i performs some of t_j's tasks on its behalf	Perform relation between things and tasks
	Negative (−)	t_i delegates some of its tasks to t_j for performance on top of performing its own tasks	Work-for relation between things and tasks
Belief	Positive (+)	t_i has beliefs that affirm some of t_j's beliefs	Affirm relation between things' beliefs
	Negative (−)	t_i has beliefs that contradict some of t_j's beliefs, which makes t_j amend its beliefs	Contradict relation between things' beliefs

An illustration from Decker and Lesser's work is when an agent determines that executing a local activity would *facilitate* the work of a peer. As a result, this agent prioritizes the execution of this local activity in support of this peer. Other relations include *cancel*, *inhibit*, *constrain*, *enable*, and *cause*.

Thanks to the GBTR framework, we reason over things' goals depending on whether the impact of the goal-driven influence relation is positive or negative on things. We recall that a goal-driven influence relation triggers the formation of both task-driven influence relation and resource-driven influence relation. Beliefs are excluded from the reasoning since belief-driven influence relation does not trigger the formation of any relation. To illustrate the reasoning over goals, we assume two things, t_1 and t_2, having each a respective set of goals, $t1_{\{\text{goal}_{1i}\}}$ and $t2_{\{\text{goal}_{2j}\}}$, that they are pursuing, respectively.

3.3.1 | Case of positive influence between goals

This means *facilitate*($t1_{\text{goal}_{1i}}, t2_{\text{goal}_{2j}}$) relation is formed where $t1_{\text{goal}_{1i}} \in t1_{\{\text{goal}_{1i}\}}$ and $t2_{\text{goal}_{2j}} \in t2_{\{\text{goal}_{2j}\}}$. Because of this relation, the following occurs in compliance with Table 1:

- For t_1 to achieve $t2_{\text{goal}_{2j}}$, t_1 will execute some of t_2's tasks, for example, $t2_{\text{task}_{2j}}$, where $t2_{\text{task}_{2j}} \in t2_{\{\text{task}_{2j}\}}$. As a result *peform*($t1, t2_{\text{goal}_{2j}}, t2_{\text{task}_{2j}}$) relation is formed.
- For t_1 to execute $t2_{\text{task}_{2j}}$, t_1 will use some of its resources, for example, $t1_{\text{resource}_{1i}}$, where $t1_{\text{resource}_{1i}} \in t1_{\{\text{resource}_{1i}\}}$. As a result *offer*($t1, t1_{\text{resource}_{1i}}, t2_{\text{goal}_{2j}}, t2_{\text{task}_{2j}}$) relation is formed.

3.3.2 | Case of negative influence between goals

This means *hinder*($t1_{\text{goal}_{1i}}, t2_{\text{goal}_{2j}}$) relation is formed where $t1_{\text{goal}_{1i}} \in t1_{\{\text{goal}_{1i}\}}$ and $t2_{\text{goal}_{2j}} \in t2_{\{\text{goal}_{2j}\}}$. Because of this relation, the following occurs in compliance with Table 1:

- For t_2 to achieve $t1_{\text{goal}_{1i}}$, t_2 will execute some of t_1's tasks, for example, $t1_{\text{task}_{1i}}$, where $t1_{\text{task}_{1i}} \in t1_{\{\text{task}_{1i}\}}$. As a result *workfor*($t2, t1_{\text{goal}_{1i}}, t1_{\text{task}_{1i}}$) relation is formed.
- For t_2 to execute $t1_{\text{task}_{1i}}$, t_2 will use some of its resources, for example, $t2_{\text{resource}_{2j}}$, where $t2_{\text{resource}_{2j}} \in t2_{\{\text{resource}_{2ji}\}}$. As a result *taveover*($t2, t2_{\text{resource}_{2j}}, t1_{\text{goal}_{1i}}, t1_{\text{task}_{1i}}$) relation is formed.

3.4 | How to put social things into action?

Figure 6 is our approach that actions Plug&Play social things. The approach runs over two stages, design-time (d) and run-time (r), featuring each multiple modules that would execute necessary operations.

3.4.1 | Design-time stage

IoT engineers proceed with forming the necessary social relations between things that are identified (or detected) in the under-consideration ecosystem such as, the smart elderly care-center. To this end, the IoT engineers initiate the connection module that analyzes the relevant things according to their functionalities (what they do) and complimentary, antagonism, and competition social-relations (1d). This module produces the relevant networks of things; it is not a must that all the social relations between things exist nor that each thing must be part of a network. Some things in the ecosystem could remain independent from the networks.

3.4.2 | Run-time stage

Users proceed with submitting their needs/requirements using the definition module that takes two inputs to

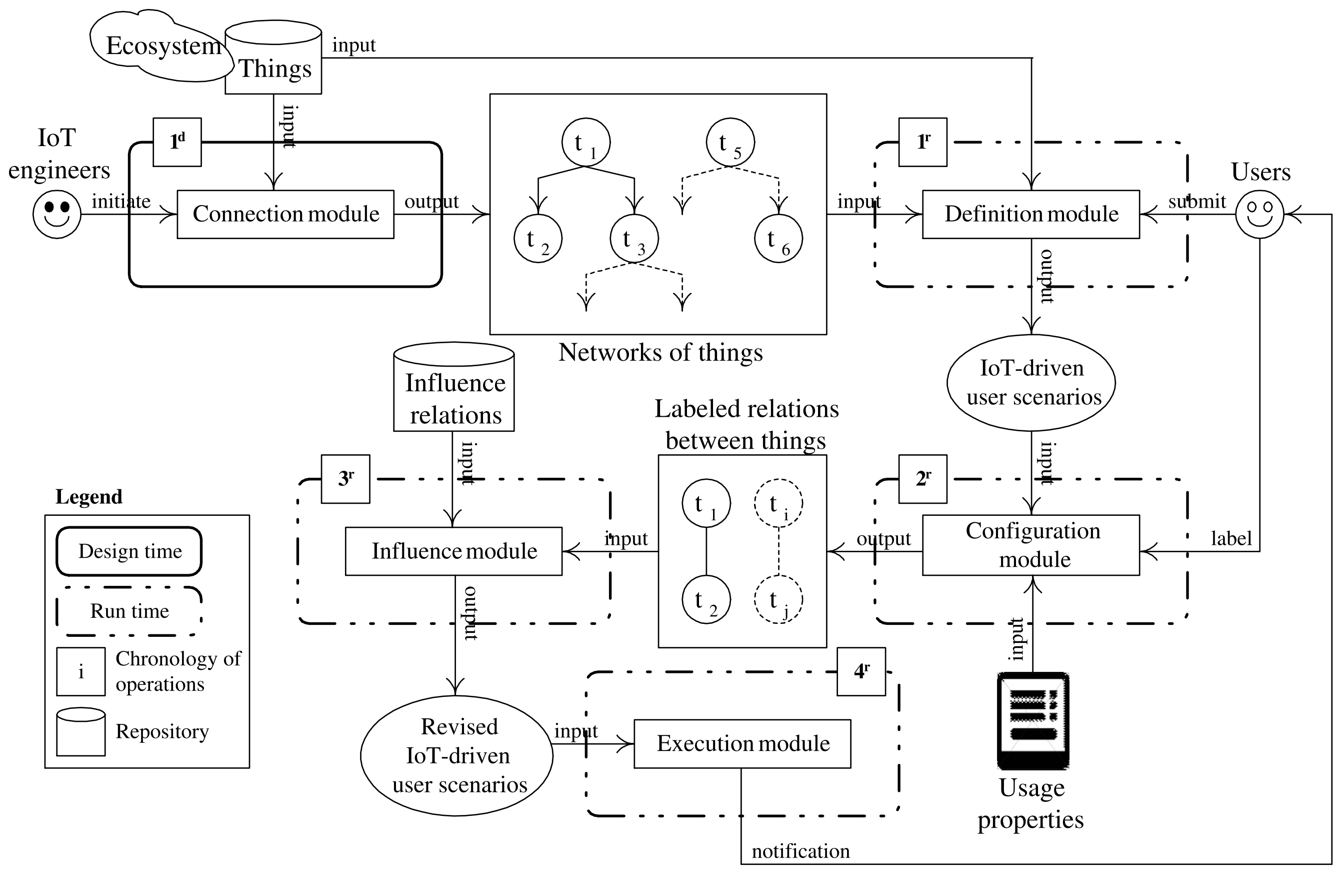


FIGURE 6 Approach in support of the functioning of Plug&Play social things

produce what we here refer to as IoT-driven user scenarios[5] (1r). These inputs are the networks of things and repository of things. In terms of scenarios, an example could be setting up the movie-projection room for the smart elderly care-center's patients. As stated in Section 3.1, the networks of things are useful when it comes to including more things in scenarios thanks to the complimentary relation, selecting things before inclusion in scenarios thanks to the competition relation, and comparing things once included in scenarios thanks to the antagonism relation. Once an IoT-driven user scenario's things are all known in terms of who will do what, where, and when, the users continue with labeling the social relations that exist among particular things in these scenarios (2r). To this end, the users initiate the configuration module and rely on the list of usage properties namely ul, li, lr, le, so, and mo. The result is a set of labelled social relations allowing to control these relations' usage cycles as described in Section 3.2. For instance, the complimentary relation between the TV and the remote control is only valid in the projection room. After this step, the influence module analyses potential impacts between things using the repository of influence relations between things that is structured as per Table 1 (3r). As discussed in Section 3.3, impacts could target goals that things will pursue, beliefs that things will refer to, tasks that things will perform, and finally resources that things will consume. It is expected that the influence module produces a revised IoT-driven user scenario since, some things could end-up revising their goals while some could end-up offering their resources, for example, Finally, the revised IoT-driven user scenarios are submitted to the execution module for completion along with keeping users notified about the progress of this completion (4r).

To demonstrate the mix of social relations, usage properties of social relations, and influence relations, we discuss two scenarios focussing on complimentary and antagonism social-relations, respectively. Competition social-relation is not considered since it does not impact the influence analysis between things. This relation is relevant for thing selection during the development of IoT-driven user scenarios.

3.4.3 | Scenario 1: complimentary social-relation

Scenario one is to convert the dining room in the smart elderly care-center into a projection room where a movie will be projected on a certain day and time. As per Figure 6-1r, a sample of things (t_i) participating in this scenario are one TV (t_1), four sofas ($\{t_2\}$), two side tables ($\{t_3\}$), and three standing lights ($\{t_4\}$). Noticing that the remote control is not among the

[5] *Scenario specification does not fall into the scope of the current paper. Works on process-of-things [18] and thing-aware business processes [35] could be adopted for this specification.*

participating things, the TV 'steps in' by recommending to the event organizers to include the remote control (t_5) in the setup of the projection room based on the complimentary relation that the TV has with a particular remote control in the network of complimentary things. Should the organizers accept the TV's recommendation that will be reflected on Equation (1), the organizers proceed as follows:

1. As per Figure 6-2r, the organizers assign a usage property to the complimentary relation, as they see fit. For instance, limited and single-occurrence seem appropriate allowing to restrict first, this relation between the TV and remote control to the specified day, time, and venue and second, the number of complimentary relations that could concurrently be enabled. Indeed, the rest of things like sofas and side tables could also recommend their own peers for inclusion in the projection room.
2. As per Figure 6-3^{r}, the organizers check whether the remote control (t_5) has any influence relation with the rest of things (t_{1-4}). To this end, the following takes place with focus on goals, only. Since the remote control has its own goals ($t_{5_{\{g_{5i}\}}}$, e.g. *adjust volume*), Algorithm 1 checks if either facilitate relation or hinder relation exists between the remote's goals and other things' goals ($t_{1,2,3,4\{g_{1j,2j,3j,4j}\}}$). Should one of these relations exist, we consider that there is either a positive influence or a negative influence between things at the goal level. This influence also means forming relations between things and goals, between things and tasks, and between things and resources. All this happens in compliance with the GBTR framework. Otherwise, we consider that there is a neutral influence between things.

Algorithm 1 Goal influence detection between things

```
Input Set of things' goals $\mathcal{T}_{\mathcal{G}}\langle t_{1_{\{goal_{1j}\}}}, ..., t_{i_{\{goal_{ik}\}}}\rangle$
Input Set of influence relations between goals $\mathcal{I}_{\mathcal{G}}\langle h(goal_1, goal_2), ..., f(goal_i, goal_j)\rangle$
                                              ▷ $h$ for hinder and $f$ for facilitate
Output Set of goal-driven influence relations between things ($t_i$, $t_j$, goal:$inflType$)
                                ▷ $inflType$ is either positive, negative, or neutral
for all $t_i \in \mathcal{T}_{\mathcal{G}}$ do
    $goal_{ij} = extract(t_{i_{\{goal_{ij}\}}})$  ▷ $extract$ function collects a thing's goals
    for all $t_{i+1} \in \mathcal{T}_{\mathcal{G}}$ do
        $goal_{(i+1)k} = extract(t_{(i+1)_{\{goal_{(i+1)k}\}}})$
        if $\exists(goal_{ij}, goal_{(i+1)k}) \in \mathcal{I}_{\mathcal{G}}$ then
            $set(t_i, t_{i+1}, goal{:}positive \oplus goal{:}negative)$  ▷ set relations between goals
                                          ▷ set relations between things and tasks
                                      ▷ set relations between things and resources
        else $set(t_i, t_{i+1}, goal{:}neutral)$
```

3.4.4 | Scenario 2: antagonism social-relation

Building upon scenario 1, scenario two targets similar things like sofas ($\{t_2\}$), side tables ($\{t_3\}$), and lights ($\{t_4\}$) that are part of the projection room. For illustration, some sofas are leather made while others are fabric made offering each a different level of comfort. Thus, a leather sofa (t_2^L) and a fabric sofa (t_2^F) are connected through an antagonism social-relation which will be reflected on Equation (2). By analogy to Scenario 1, the organizers proceed as follows:

1. As per Figure 6-2r, the organizers assign a usage property to the antagonism relation, as they see fit. For instance, le seems appropriate allowing to restrict this relation between the leather sofa and fabric sofa to the specified projection room and also to extend this relation to other rooms, should the initial projection room turn out inappropriate because of the unexpected high number of guests.
2. As per Figure 6-3r, the organizers check whether the leather sofa (t_2^L) has any influence relation with the fabric sofa (t_2^F). To this end, the following takes place with focus on beliefs.

Since, the leather sofa has its own beliefs ($t^{L}_{2_{\{belief_{2i}\}}}$, e.g. *sofa is 1-year old*), Algorithm 2 checks if either affirm relation or contradict relation exists between the leather sofa's beliefs and the fabric leather's beliefs ($t^{F}_{2_{\{belief_{2j}\}}}$). Should one of these relations exist, we consider that there is either a positive influence or a negative influence between things at the belief level, which could impact the consistency of beliefs. All this happens in compliance with the GBTR framework. Otherwise, we consider that there is a neutral influence between things.

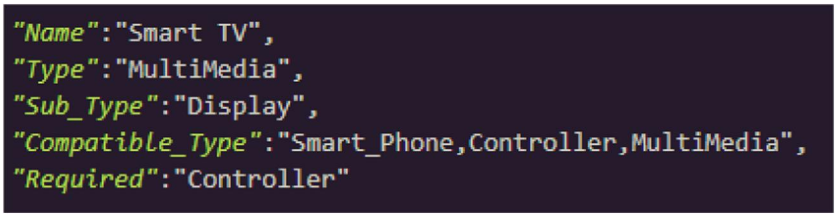
Algorithm 2 Belief influence detection between things

```
Input Set of things' beliefs $\mathcal{T}_{\mathcal{B}}\langle t_{1_{\{belief_{1j}\}}}, ..., t_{i_{\{belief_{ik}\}}}\rangle$
Input Set of influence relations between beliefs $\mathcal{I}_{\mathcal{B}}\langle a(beliefs_1, beliefs_2), ..., c(beliefs_i, beliefs_j)\rangle$  ▷ $a$ for affirm and $c$ for contradict
Output Set of thing influence relations between things ($t_i$, $t_j$, belief:$inflType$)
                                ▷ $inflType$ is either positive, negative, or neutral
for all $t_i \in \mathcal{T}_{\mathcal{B}}$ do
    $belief_{ij} = extract(t_{i_{\{belief_{ij}\}}})$        ▷ $extract$ function collects a thing's beliefs
    for all $t_{i+1} \in \mathcal{T}_{\mathcal{B}}$ do
        $belief_{(i+1)j} = extract(t_{(i+1)_{\{belief_{(i+1)j}\}}})$
        if $\exists(belief_{ij}, belief_{(i+1)k}) \in \mathcal{I}_{\mathcal{B}}$ then
            $set(t_i, t_{i+1}, belief{:}positive \oplus belief{:}negative)$  ▷ set relations between beliefs
        else $set(t_i, t_{i+1}, belief{:}neutral)$
```

3.5 | Demonstration

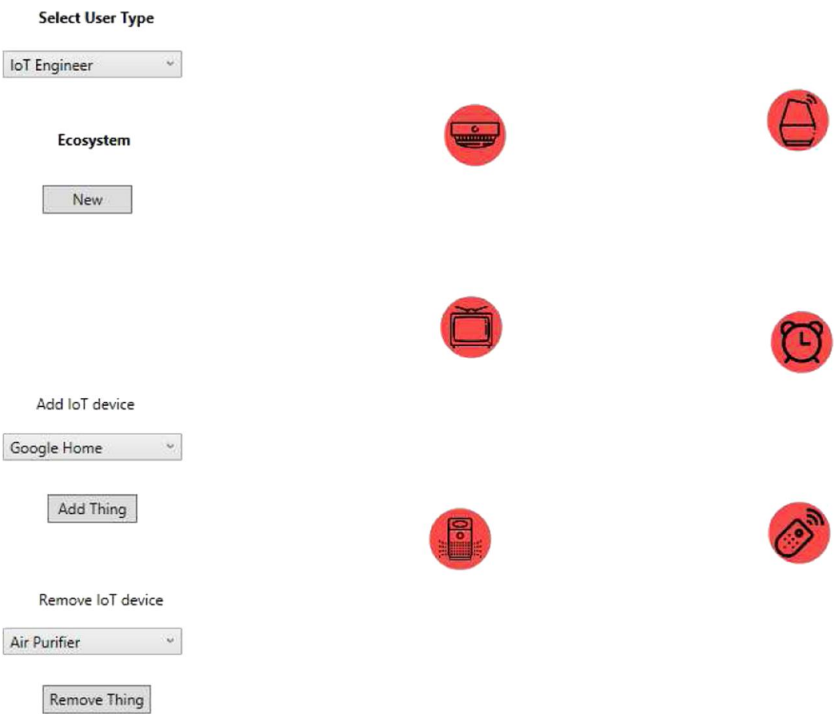
A video illustrating how we put Plug&Play social things into action is available at http://social.connect.rs/demo.mp4. We created a custom set of things allowing the simulation of complimentary, antagonism, and competition relations between these things. Figure 7 is an excerpt of the properties of an IoT device that is, in our case, smart TV. The properties are *name* holding a value like smart TV and remote control, *type* being the category like controller and home appliance to which the IoT device belongs, *sub-type* refining category into sub-

FIGURE 7 Excerpt of an Internet of Things device description



```
"Name":"Smart TV",
"Type":"MultiMedia",
"Sub_Type":"Display",
"Compatible_Type":"Smart_Phone,Controller,MultiMedia",
"Required":"Controller"
```

FIGURE 8 Drag&Drop feature during device selection



Select User Type

IoT Engineer

Ecosystem

Add IoT device

Google Home

Remove IoT device

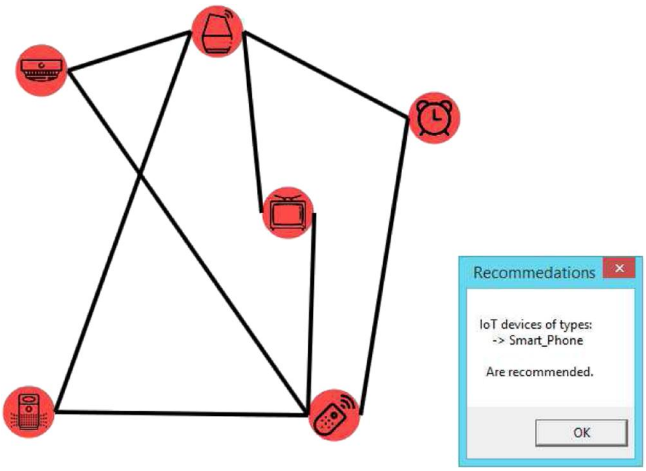
Air Purifier

categories, *compatible-type* holding the lists of all peers' types that could connect to the IoT device, and *required* referring to the category that must exist in the IOT ecosystem so, that, the under-analysis IOT device can provide its functionalities.

The system was developed in-line with the approach presented in Figure 6. We implemented the modules in C# and necessary graphical user interfaces on Unity (unit.com). Using the connection module, the IoT engineer drags&drops the IoT devices into the working space (Figure 8). In the current case, the IOT engineer has chosen the following devices: Smart TV, Google Home, Remote Control, Alarm Clock, Air Purifier, and Smoke Detector. The IoT engineer aims at benefiting from these devices, that co-exist in the same room, through the definition of one or many IoT systems. Next, the simulation begins by determining the social relations that exist between each couple of IoT devices. These social relations are used to establish one or many networks of compatible devices (some could remain independent depending on the selected ones and their attributes). Once the networks are set, the system generates an adjacency matrix between devices (Figure 9). This matrix is made by comparing *compatible-type* property of one device to the *type* property of another device and vice versa (to ensure a 2-way compatibility). Once device compatibility is confirmed, the initial values of the social relation's tuple < id,

$$\begin{array}{c|ccc} & TV & RC & \dots \\ \hline TV & 0 & 1 & \dots \\ RC & 1 & 0 & \dots \\ \dots & \dots & \dots & \dots \end{array}$$

FIGURE 9 Representation of the adjacency matrix

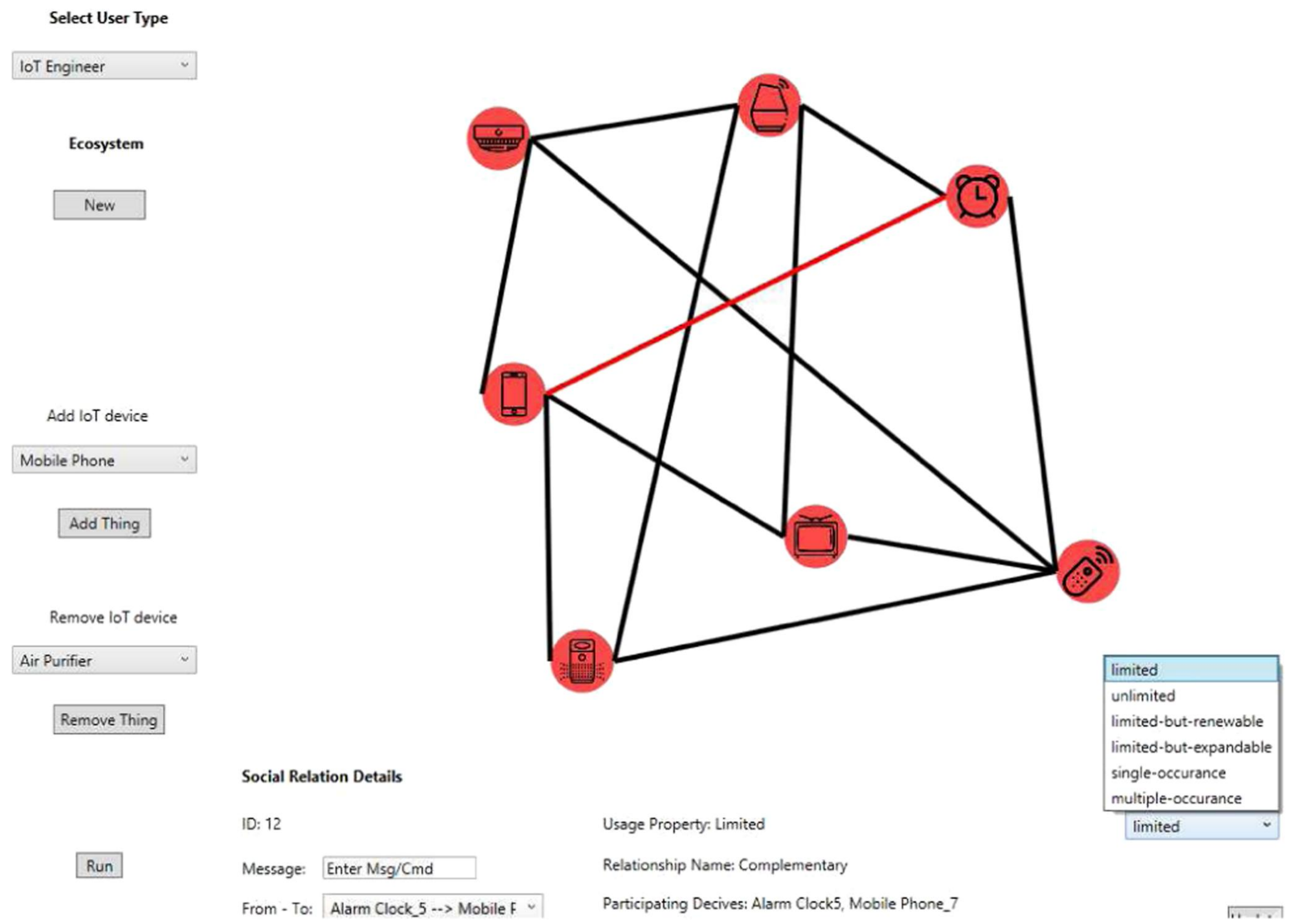


Recommedations

IoT devices of types:
-> Smart_Phone

Are recommended.

FIGURE 10 Complimentary relations between things. IoT, Internet of Things



Select User Type: IoT Engineer

Social Relation Details

ID: 12

Message: Enter Msg/Cmd

From - To: Alarm Clock_5 --> Mobile F

Usage Property: Limited

Relationship Name: Complementary

Participating Decives: Alarm Clock5, Mobile Phone_7

FIGURE 11 Forming social relations between devices

name, P, up, UC_{up}> are set (name for relation's name, P for participating devices, up for usage property, and UC_{up} for the usage cycles).

Using the definition module, the IoT engineer can load three scenarios: complimentary (where devices are recommended on the basis of a device's requirements), antagonism (where devices are compared to each other), and competition (where conflicting devices are removed). In the current case, and based on the selected devices, there are neither antagonism no competition relations. However, a mobile phone was recommended to the IoT engineer, thanks to the complimentary relation(s) (Figure 10). Indeed, the system examines the types of devices that are required by the already selected devices, for example, air purifier requires a mobile phone. Once the IoT engineer accepts the recommendation by adding the recommended devices, the system proceeds with forming new social relations between devices (Figure 11).

The configuration module focusses on the usage properties of the social relations. It allows the IoT engineer to modify the values of some properties and test the connection between devices (Figure 11). In addition, as the system proposes all possible social relations between the selected devices, the IoT engineer could consider some relations as necessary and others as not relevant in the current case study. For instance, in this example the IoT engineer decides to create two IoT ecosystems: the first would include smart TV, Google home, remote control, and alarm clock, while the second would include air purifier, smoke detector, and mobile phone. As such, any social relation between devices that do not belong to the same IoT ecosystem will not be exploited. Contrarily, these social relations could be used in other situations like using the social relation between smoke detector and remote control in another context and/or to define another IoT ecosystem.

4 | CONCLUSION

This article presented an approach to model and simulate Plug&Play social things that would collaborate to satisfy users' complex needs. Social because these things are members of networks so they ‘crawl’ through these networks to identify with whom they could collaborate, with whom they would compete, and who could recommend them. The networks proposed in this article are built-upon three relations referred to as complimentary, antagonism, and competition. To ensure proper use of these relations, usage properties have been defined and referred to as li, lr/le, and so/mo. Usage properties permit to declare when and where social relations are enabled, for how long, and for how many times. When social things participate in user-driven situations, they influence each other either positively or negatively, which could impact the goals they were pursuing, the beliefs they were managing, the tasks they were performing, and the resources they were consuming. Combining social relations between things, usage properties of social relations, and influence relations between things has been coordinated through four stages that are connecting to demystify social relations between things, influencing to examine the impact of social relations on things, playing to make things perform while considering influence, and incentivising to reward things based on their performance. A system developed in the context of smart elderly care-centres demonstrated the technical doability of the approach for modelling and simulating Plug&Play social things. The system is an IDE allowing IoT engineers to orchestrate the work of several social things. In term of future work, we would like to examine the scalability of the system when hundreds of things co-exist in the same ecosystem. Would the different social relations between things slow down the system? We would also like to the composition of social things from the orchestration and the choreography perspectives. The former would call for a central component that would 'tell' things the actions to carry out while the latter would rely on peer-to-peer interactions to 'tell' things the actions to carry out, as well. Pros and cons of each perspective in the context of social things would be examined.

ORCID

Thar Baker https://orcid.org/0000-0002-5166-4873

REFERENCES

1. Dourish, P., Bell, G.: Divining a digital future: mess and mythology in ubiquitous computing. The MIT Press, Cambridge (2011)
2. Middleton, J.: Long live the thing! temporal ubiquity in a smart vintage wardrobe, Ubiquity. The Journal of Pervasive Media. 1(1), 7–22 (2012)
3. Atzori, L., et al.: The social Internet of things (SIoT) - when social networks meet the Internet of things: concept, architecture and network characterization. Comput. Network. 56(16), 3594–3608 (2012)
4. Ortiz, A.M., et al.: The cluster between Internet of things and social networks: review and research challenges. IEEE Internet Things J. 1(3), 206–215 (2014)
5. Marche C., et al.: How to exploit the Social Internet of Things: Query Generation Model and Device Profiles' Dataset. Computer Networks 174, 1–34 (2020)
6. Maamar, Z., et al.: Commitments to regulate social web services operation. IEEE Trans. Serv. Comput. 7(2), 154–167 (2014)
7. Maamar, Z., et al.: LinkedWS: a novel web services discovery model based on the metaphor of "social networks". In: Simulation Modelling Practice and Theory, 19(10). Elsevier Science Publisher (2011)
8. Brambilla, M., Fraternali, P., Vaca, C.: A notation for supporting social business process modeling. In: Proceedings of the Fourth Workshop on Business Process Management and Social Software (BPMS2' 2011) held in conjunction with the Seventh International Conference on Business Process Management (BPM'2011), Lucerne (2011)
9. Yahya, F., et al.: A fuzzy-based approach for identifying candidate business processes for socialization. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 27(1), 19–39 (2019)
10. Zhang, C., Cheng, C., Ji, Y.: Architecture design for social web of things. In: Proceedings of the 1st International Workshop on Context Discovery and Data Mining (ContextDD'2012), Beijing (2012)
11. Elnaffar, S., Maamar, Z., Sheng, Q.Z.: When clouds start socializing: the sky model. Int. J. E Bus. Res. 9(2), 1–7 (2013)
12. Kott, A., Alberts, D.S.: How do you command an army of intelligent things? IEEE Computer. 50(12), 96–100 (2017)
13. Chen, J., et al.: WaaS: wisdom as a service. IEEE Intell. Syst. 29(6), 40–47 (2014)
14. Katasonov, A., et al.: Smart semantic middleware for the Internet of things. In: Proceedings of the 5th International Conference on Informatics in Control, Automation and Robotics (ICINCO'2008), Funchal (2008)
15. Mzahm, A.M., Ahmad, M.S., Tang, A.Y.C.: Agents of things (AoT): an intelligent operational concept of the Internet of things (IoT). In: Proceedings of the 13th International Conference on Intellient Systems Design and Applications (ISDA'2013), Bangi (2013)
16. Savaglio, C., et al.: Agent-based computing in the Internet of things: a survey. In: Proceedings of the 11th International Symposium on Intelligent Distributed Computing (IDC), Belgrade (2017)
17. Kwan, J., et al.: An agentified use of the internet of things. In: Proceedings of the 2016 IEEE International Conference on Internet of Things (iThings'2016) and IEEE Green Computing and Communications (GreenCom'2016) and IEEE Cyber, Physical and Social Computing (CPSCom'2016) and IEEE Smart Data (SmartData'2016), Chengdu (2016)
18. Maamar, Z., et al.: Storytelling integration of the internet of things into business processes. In: Proceedings of the Business Process Management Forum (BPM Forum'2018) held in conjunction with the 16th International Conference on Business Process Management (BPM'2018), Sydney (2018)
19. Shen, Z., et al.: Dynamic generation of Internet of things organizational structures through evolutionary computing. IEEE Internet Things J. 5(2), 943–954 (2018)
20. Maamar Z., et al.: Network-based social coordination of business processes. Information Systems. 58, 56–74 (2016)
21. Maamar, Z., Sellami, M., Masmoudi, F.: A transactional approach to enforce resource availabilities: application to cloud computing. College of Technological Innovation, Zayed University, Dubai (2020)
22. Friedemann, M., Christian, F.: From the Internet of computers to the Internet of things. In: From Active Data Management to Event-Based Systems and More – Papers in Honor of Alejandro Buchmann on the Occasion of his 60th Birthday. Springer (2010)
23. Porter, M.E., Heppelmann, J.E.: How smart, connected products are transforming competition. Harv. Bus. Rev. 92(11), 1–23 (2014)
24. Weber, R.H., Weber, R.: Internet of things – legal perspectives. Springer-Verlag Berlin Heidelberg (2010)
25. Pticek, M., Podobnik, V., Jezic, G.: Beyond the Internet of things: the social networking of machines. Int. J. Distrib. Sens. Netw. 12(6), 1–15 (2016)
26. Thangavel, G., Memedi, M., Hedström, K.: A systematic review of social Internet of things: concepts and application areas. In: Proceedings of the 25th Americas Conference on Information Systems (AMCIS'2019), Cancun (2019)
27. Militano L., et al.: Enhancing the navigability in a social network of smart objects: A Shapley-value based approach. Computer Networks. 103, 1–14 (2016)
28. Hussain, N., et al.: Software agent-centric semantic social network for cyber-physical interaction and collaboration. Int. J. Softw. Eng. Knowl. Eng. 30(6), 859–893 (2020)
29. Roopa, M.S., et al.: Social Internet of things (SIoT): foundations, thrust areas, systematic review and future directions. Comput. Commun. 139, 32–57 (2019)
30. Maamar, Z., Hacid, H., Hunhs, M.N.: Why web services need social networks. IEEE Internet Comput. 15(2), 90–94 (2011)
31. Boukadi K., et al.: Norm-based and commitment-driven agentification of the Internet of Things. Internet of Things 6, 100042 (2019)
32. Maamar, Z., Akhter, F.: How to deal with influence in multiagent systems? In: Proceedings of the 2001 IEEE Systems, Man, and Cybernetics

Conference: e-Systems and e-Man for Cybernetics in Cyberspace, Tucson (2001)

33. Castelfranchi, C., Conte, R.: Distributed artificial intelligence and social science: critical issues. In: O'Hare, G.M.P., Jennings, N.R. (eds.) Foundations of Distributed Artificial Intelligence. John Wiley & Sons, New York, NY (1996)
34. Decker, K.: Lesser, V.R.: Generalizing the partial global planning algorithm. Int. J. Coop. Inf. Syst. 1(2), 319–346 (1992)
35. Suri, K., et al.: Semantic framework for Internet of things-aware business process development. In: Proceedings of the 26th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'2017), (2017). https://doi.org/10.1109/WETICE.2017.54

How to cite this article: Maamar Z, Sellami M, Masmoudi F, et al. A Plug&Play approach for modeling and simulating applications in the era of internet of social things. *IET Smart Cities*. 2021;3:29–40. https://doi.org/10.1049/smc2.12005