IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received August 5, 2017, accepted October 8, 2017, date of publication October 27, 2017, date of current version November 14, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2765746

Principles to Design Smart Physical Objects as Adaptive Recommenders

FEDERICA CENA[1], LUCA CONSOLE[1], ASSUNTA MATASSA[1], AND ILARIA TORRE[2]
[1]Dipartimento di Informatica, University of Torino, I-10149 Torino, Italy
[2]Dip. di Informatica, Bioingegneria, Robotica, Ingegneria dei Sistemi, University of Genova, 16145 Genova, Italy

Corresponding author: Ilaria Torre (ilaria.torre@unige.it)

ABSTRACT Recommenders have proven to be useful means to support people in their activities and in making decisions. They evolved from online recommenders to context-aware and ubiquitous recommenders. Moving forward along this line, this paper introduces the new emerging class of smart physical recommenders: context-aware recommender systems that are embedded into physical everyday objects. This paper describes the features of these systems and presents a conceptual model to design them, by analyzing a number of issues that have to be addressed by a designer and discussing the consequences of different design choices with their impact on the smartness of the designed object. The model is structured in a number of layers corresponding to different conceptual design phases in which different requirements are analyzed. The contribution of this paper is to discuss and provide design guidelines for a new rising class of recommenders that combine the features of intelligent agents, cyber-physical objects, and recommender-support systems. The description of the model is complemented by an exemplary analysis of its application.

INDEX TERMS Adaptive systems, context-aware recommenders, multi-layered design, smart objects.

I. INTRODUCTION

The design of smart objects attracted a lot of attention over the last decade and a number of examples are available in the literature [25], [34] and on the market. Thus, we believe that it is time for reflecting on the different forms of smartness these objects can exhibit and for analyzing the design patterns for modeling them. Nowadays, the label ''smart'' is used for heterogeneous objects having different levels of complexity, awareness and autonomy, and providing different types of support to people [28], [36].

In a previous paper [11] we introduced a classification of types of intelligence in smart physical objects, discussing how they can be taken into account when designing an object. The discussion of the paper was general in the sense that we did not make any assumption about the type of object to be designed and on the task and type of support to people the object could provide. In this paper we take a different point of view. We focus on a specific task – interactive adaptive support and recommendation to people — and we analyze in detail the design of smart objects performing this task. The result of this analysis is a conceptual model which is specifically suited to design smart objects performing adaptive recommendation.

Let us start by setting up the scene with some preliminary definitions of the acronyms and terms used in the paper.

An **SPO** is a **Smart Physical Object** with interacting and problem-solving capabilities, defined as the tight and seamless integration of a physical and a digital counterpart which augment each other into a unique peculiar entity. This remark is important in characterizing ''intelligence'' in the sense that intelligence cannot be independent of the physical nature of the object and must augment this physical dimension in the same way as the physical dimension is the handle to support intelligent behavior enabled by the digital dimension. Although a smart object can be seen as an ''Intelligent Agent'' according to classical definitions [52], it is different from a mere software agent since it has a body and this may impact on cognition and behavior.

In this paper our focus is on **ARSPO**: **Adaptive Recommender SPO**. Recommenders are systems that support a user in her activity or decision-making process by suggesting items (services or information) which are supposed to be relevant to her [3], [46], [47].

Recommender systems may take into account the user/s preferences and features, stored in a user profile or in a *User Model* [6], [7], [19], and may also take into account the specific context in which the user and the object interact, including aspects such as the place, the time, the weather and the presence of other objects [2]. This includes the notion of context and situation awareness [1], [5], commonly used in the literature. We use the term ARSPO to refer to an SPO which takes into account all of these features when interacting with a user.

The model we propose is centered around the idea that intelligence is multi-faceted and involves different aspects, ranging from interaction abilities to the ability of managing knowledge and of reasoning and learning from experience. All these dimensions assume peculiar features when attached to a physical object and this is the principle guiding our analysis which borrows and integrates concepts from different areas of artificial intelligence, cognitive sciences and human-computer interaction.

The model is organized in a number of layers (or steps). Isolating layers can in our view facilitate the design process and the mapping between requirements and choices in the object to be designed.

The paper is structured as follows. Section II introduces the first steps in the process of designing an ARSPO, Section III provides an overview of the layers of our model; Sections IV to VIII analyze each layer and Section IX combines the layers providing a multi-faceted approach for the design and classification of ARSPOs. Section X exploits the model to analyze a couple of ARSPOs. Section XI analyzes related work while Section XII concludes the paper.

II. PRELIMINARY DESIGN STEP: PURPOSE, TARGET USERS, OBJECT

In this section we discuss some steps that are preliminary to the design of an ARSPO and that contribute to defining the requirements for the object to be designed.

1) Identifying the purposes of the adaptive recommendation process, the target users and the type of recommendation to be provided.
2) Selecting the object to be transformed into a smart one.

A. IDENTIFYING PURPOSES AND TARGET USERS

We adopt a schematic view of context-aware adaptive recommendation as a mapping:

$$itself \times cxt \times objs \times user \rightarrow Service \times InteractionMode \tag{1}$$

where *itself* concerns features of the ARSPO, *cxt* is the context in which the process is taking place, *objs* is the presence of other smart objects, *user* is the relevant set of user features and *Service* and *InteractionMode* are the two targets of adaptive recommendation: what to recommend and how, given the user, the context, and the presence or absence of other smart objects. Designing an ARSPO involves making choices for defining the mapping and its constituents. At least the following aspects should be considered in the design (as explained in [31] and [49]):

- *Who.* Which aspects of the *user* are relevant for recommendation, i.e. for selecting the *Service* and the *InteractionMode*;
- *What.* What is the *Service* that can be provided.
- *Why.* Why does the user need to be supported.
- *When - Where.* Which aspects of the context (*cxt*), including other objects (*objs*), are relevant for predicting the items relevant for the user.

B. SELECTING THE OBJECT

The designer should select the physical object to be enhanced for providing personalised services to the user. This selection may depend on many factors that are outside the scope of this paper. The only relevant issue regards interaction; the object should be suitable for the service it offers, in the context in which the object will be used. This will have an impact on the interaction layer discussed in the following (section IV).

- *How.* How recommendation can be provided, i.e., the appropriate *InteractionMode*.

Now, we will provide examples of the high level decisions mentioned above.

Example A (A Smart Chair):

1. Identifying purposes and target users

WHO: workers at office desks

WHAT: providing recommendation for improving the ergonomic position at work

WHY: the user should be supported in order to guarantee a safe working position

WHERE: at the office desk

WHEN: during working time and specifically whenever she is in a bad position.

2. Selecting the object: a chair

HOW: the chair could perform actions (e.g., change the seatback position) or provide feedback: physical (e.g., stimuli to the user to invite her to stand) or digital (e.g., messages to the user's personal devices). This smart object will be used as a running example in the rest of the paper.

Example B (A Smart Wristband):

1. Identifying purposes and target users

WHO: a generic person

WHAT: providing recommendation for improving the user's physical activity

WHY: the user should be supported in order to decrease her level of sedentary

WHERE: everywhere

WHEN: in any moment of the day when she sits for a while.

2. Selecting the object: a wristband

HOW: the wristband may provide physical (vibration) or digital (message to the user's smart phone) feedback.

III. AN OVERVIEW OF LAYERS IN ARSPO DESIGN

Once the high level decisions are made and thus the requirements are available, the design of an ARSPO can be decomposed into five layers, as shown in Figure 1. The figure shows the relations among the layers, while Table 1 highlights, for each layer, the requirements (R) to be analysed (left column) and the set of methodological instruments and models (I) that can be used to carry out the design steps (right column).

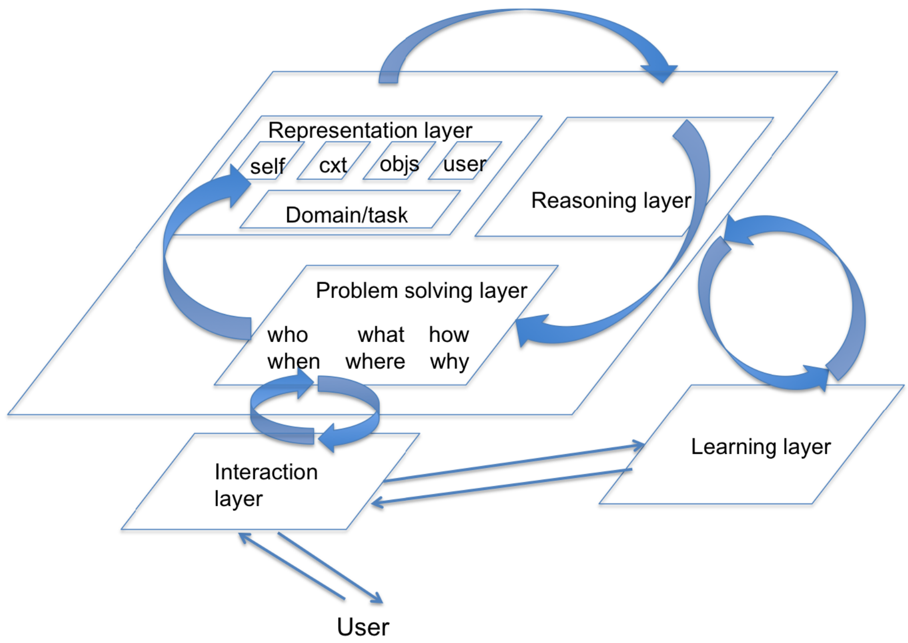

FIGURE 1. Schema of the layers.

These instruments will be adopted in the next sections to discuss each layer and are borrowed from various areas of human computer interaction and artificial intelligence. In some sense, the model can resemble in some way the BDI, Beliefs Desires Intentions model of intelligent agents [44]. We can state that the Beliefs (informational state of the agent) and the Desires (the motivational state of the agent, the objectives) of the objects are contained into our Representation (and Reasoning) layer, while the selection of the Intention (the deliberative state of the agent, i.e. what it has chosen to do) is in charge of the Problem Solving and Interaction Layers.

The next sections will present each layer, showing that a number of different **Dimensions (D)** can be modeled by using the specified tools and techniques and that different levels of sophistication, complexity and intelligence of the ARSPO can be obtained. Figure 2 provides an overview of the dimensions of analysis for each layer.

IV. INTERACTION LAYER

The design choices in this layer concern user-ARSPO interaction. As mentioned in Table 1, the type of interaction can be classified in four main levels: Task, Syntactic, Semantic and Interaction level.

This schema defines a set of steps that can guide the definition of the interaction capabilities of the ARSPO, from *static*, where the object is able to perform basic tasks, to *dynamic*, where the interaction can be modified depending on the contextual elements, the user's and the object's goals. Two dimensions influence the interaction capabilities: the level of physical-digital integration and the affordances of the object.

A. LEVELS OF INTEGRATION

The integration of digital and physical elements is the distinctive feature of an SPO, and thus of an ARSPO. Multiple sensory input/output devices, communication mechanism, processing and storage capabilities can be integrated forming a network of physical and digital elements. Unlike more traditional embedded systems, a fully integrated one defines a new entity with new sensing and interaction capabilities.

A good level of integration is fundamental to support interaction with the ARSPO at "semantic" and "interaction" levels. This is a first dimension with alternatives design decisions; the alternatives can be represented as a spectrum from objects offering a GUI-like interface (where the digital dimension is separated from the physical one), to objects offering tangible interaction (TUI), up to a fully seamless integration of the two aspects.

B. AFFORDANCES

Moving toward the top of the spectrum (that is seamless integration), a second aspect becomes relevant: the need of new types of affordances. The concept of affordance dates back to the late 70's and is defined as all the *action possibilities* latent in the environment and specifically in an object, independently of the ability of living beings to recognize them [23]. According to an interaction design perspective [32], *affordance* is the capability of an object to make evident to users what actions can be performed with it, inducing the right interaction between users and objects [42]. The more the integration is seamless, the more complex the interaction design is. This is because ARSPOs maintain their original aspect and functions but in the meantime they incorporate

TABLE 1. Design layers: requirements (R) and methodological instruments (I).

INTERACTION LAYER	
(R) Requirements and design decisions concern the integration between the physical and digital dimensions of the object and thus the interfaces it can offer to users.	(I) In order to analyze how an ARSPO interacts with the user, a useful paradigm is the Activity Theory, which is used in Human Computer Interaction to analyze how agents carry on activities, performing concrete actions and specific operations to achieve a goal. The activities are performed by using a tool, like a mediator, that can be a physical object (e.g., an hammer) or an abstract tool (e.g., a set of knowledge who influence the activities). In [11] the type of interactions between humans and artifacts are defined at the following levels: • *Task level* refers to the functionalities an object provides and the tasks reachable using such functions; • *Syntactic level* identifies the modes to interact with an ARSPO, considering both physical and digital properties; • *Semantic level* refers to the meaning associated to the interaction with an ARSPO; • *Interaction level* constitutes the dialogue and relationships between a user, an artifact and the environment. It represents the highest level of interaction, based on complex forms of influence between people and objects.
REPRESENTATION LAYER	
(R) Requirements and design decisions regard the type of information that the object should manage. At an abstract level they can be divided in knowledge about: (i) itself, (ii) context, (iii) other smart objects, (iv) user, (v) domain, including the services the ARSPO can offer, the interaction modes it can choose and the criteria for making these choices.	(I) Methodological instruments to manage the representation layer concern at least two issues: - *level of conceptualization* (i.e., representation of the features and properties of the users, other objects, items to be recommended, the environment, etc.). The design choices range from no conceptualization at all (with features and properties represented only as labels or names) to a full conceptualization with ontologies [24], with a full spectrum of intermediate situations. - *ontological categories to model knowledge*: a useful model can be borrowed from model-based reasoning [26]. It describes four types of knowledge [14]: • *Teleology* (the goal(s) an object is designed to have), • *Function* (the functions the object can perform to reach its goal(s), “what the object can do"), • *Behaviour* (the way a specific function is executed, i.e., the actions that the object performs when a function is activated, “how to do"), • *Structure* (the physical/digital structure of the object).
REASONING LAYER	
(R) Requirements and design decision concern the ARSPO's ability to make inferences on information available from the context, the user, other objects, exploiting knowledge in the representation layer.	(I) Different forms of reasoning can be classified schematically by referring to a limited set of reasoning patterns commonly used in AI applications [48]: (i) *deductive*, (ii) *abductive*, (iii) *inductive*, (iv) *analogical*, (v) *common sense*. Deduction concerns inferences that extend factual information using the principle of logical consequence. Abduction allows explaining observations using the available knowledge and is a form of non monotonic reasoning. Basically, it corresponds to “inverting" deduction: while from $a \rightarrow b$ and a one may deduce b; from $a \rightarrow b$ and b one may abduce a as an explanation of b. Inductive inference allows an object to generalize starting from a set of observations to draw general relations. Over-simplifying using the example above, induction may lead to infer the general rule $a \rightarrow b$ from a repeated simultaneous observation of a and b and is obviously another form of defeasible reasoning (not supported, as in the case of abduction, from classical logics). By defeasible we mean that the conclusions might have to be retracted when new information is available. For example, a simultaneous observation of a and *not* b may invalidate the inductive inference above. Other forms of common-sense, non monotonic inference, correspond to drawing conclusions by default (i.e., making inferences based on what typically happens) or in absence of evidence contrary to what is concluded. For example, an object might infer by default that the person using it is its owner and behave consequently, but it might have to retract this inference and change its behaviour accordingly when it discovers that it is in use by another person. Analogy allows an object to make inferences on a specific situation based on similarities between the situation and previous ones (e.g., case-based reasoning). These two forms of reasoning are intrinsically defeasible as well.
PROBLEM SOLVING LAYER	
(R) Requirements and design decision concern the strategies to provide adaptive recommendation, i.e., “what" services should be offered, “how", “where" and “when", given the individual user (“who") and context. This is the core layer of an ARSPO.	(I) This layer shares the reasoning techniques described in the above layer, but focuses on the dimension of the services to be offered, the interaction modes and the mappings to select them.
LEARNING LAYER	
(R) Requirements and design decision regard the ARSPO's ability to learn from the interaction with the user, possibly after feedback on the recommendation process. The ability to learn has been often considered a fundamental ingredient of intelligence [48].	(I) Methodological instruments to manage the learning layer concern the following issues: • the types of input data (physical and digital data), • the output (target), in terms of what is acquired/revised/refined/updated; • the type of feedback the ARSPO can exploit in the learning task, • the methods and algorithms that the ARSPO adopts, • the modality and frequency of the learning task. Learning is a reasoning activity and thus concerns the reasoning techniques discussed above. However, learning is peculiar in the sense that not only does it exploit a knowledge base but it builds/refines/revises it as well. This is the reason why it is kept separate from the reasoning layer.

intelligent behavior and new functionalities. A smart chair that changes its form on the basis of the user's position, that alerts her to stand up and move when she is sit for a long time or even asks her if she would like to be kept warm, may change the user perception of such an object and may generate expectations which can be different from person

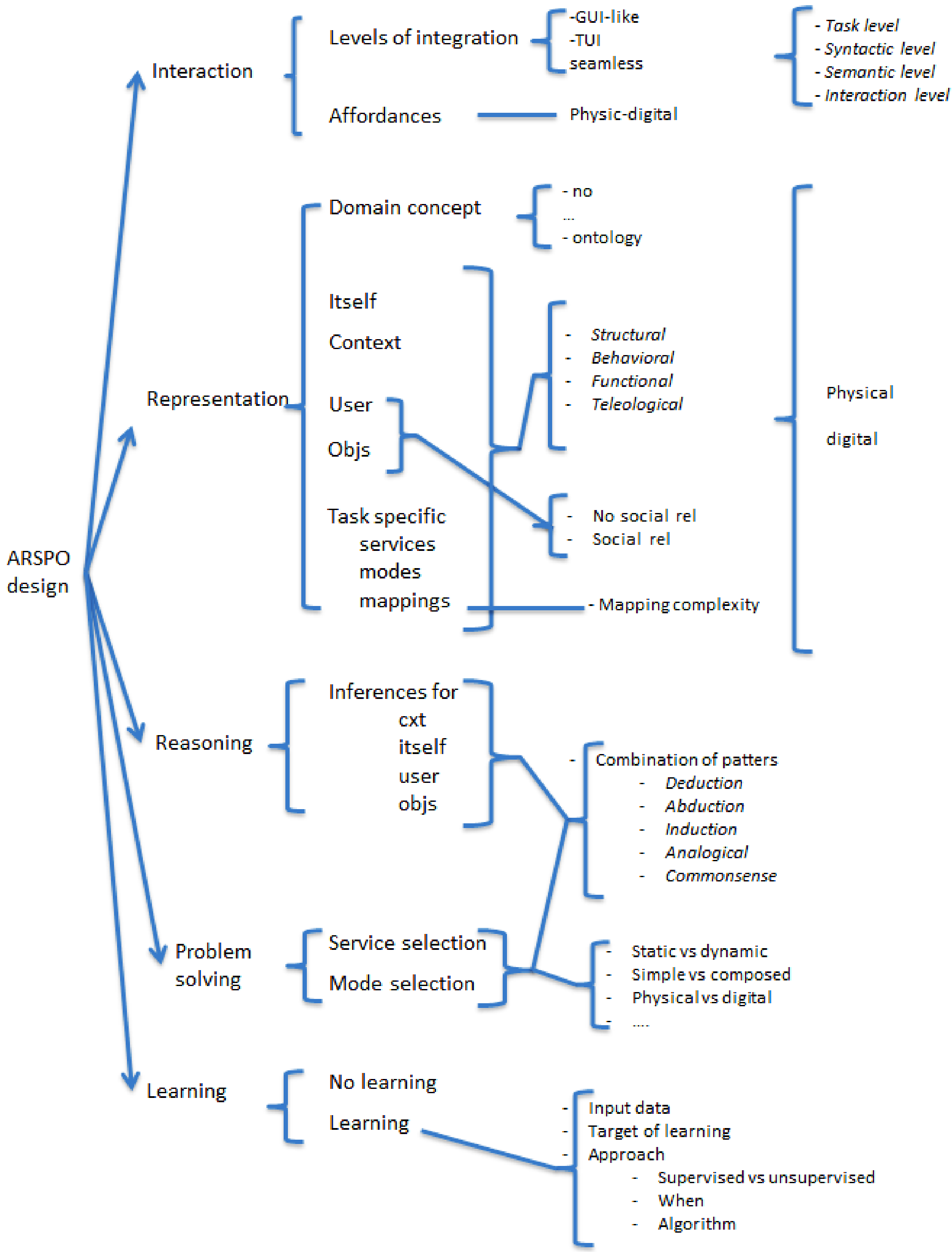

FIGURE 2. Dimensions of analysis (D).

to person. Some people could be induced to overestimate the actual abilities of the chair, others could be confused about what the chair can really do.

The design of an ARSPO should take into account their intermixed physical and digital nature, identifying new kinds of affordances and interaction handles [37], [41]. According to [22], we can analyze this issue as an input-output problem. An ARSPO is the enhancement of a physical object characterized by a shape and some interaction handles; the digital enhancement should respect the object traditional function and interaction model [13] and avoid any conflict between them. The psychological approach of cognitive dissonance [18] can be useful to analyze the distance between what people perceive to be able to do with the object and what really happens in the interaction with the object. Norman speaks of a ''gulf of interaction'', as a figurative distance between what an object suggests to do to the user and the results derived from this interaction [42].

Following these considerations, new types of affordances should be singled out, focusing on the way physical features can suggest what, when and how users can interact with an ARSPO, exploiting the capabilities provided by the cooperation of the digital and physical layers [37]. In fact, affordances in ARSPOs are more than what actions are possible through a particular object and what consequences of these actions are. They are due to the combination of what actions are possible through the physical properties and the object's teleological knowledge, which concerns the object's goals. Thus, by combining these two parts, affordances in ARSPOs put together interaction abilities and cognitive abilities, defining a novel unity of the behavioural act, specifying the goal-directed action. Each part represents just a facet of the capabilities of the objects, and both are required to truly perform an action and to achieve a goal.

C. DISCUSSION ON DESIGN CHOICES

In summary, we isolated two issues that should be taken into account during interaction design and that strongly impact the levels of perceived smartness of an ARSPO: the degree of integration of the digital and physical dimensions and the naturality the affordances offered by the ARSPO.

In both cases we have a qualitative spectrum of alternatives that can be assessed by evaluating the object performance with users. This is an important remark since this layer cannot be designed independently of the user, as typical in user-centered design approaches, and consequently there is also a high degree of subjectivity in assessing the level of smartness of the resulting ARSPO.

A key issue, however, is that the level of sophistication achieved in the design of the interaction layer is critical for characterizing the level of interaction offered to and perceived by the user. High levels of sophistication in the other layers can be vanished if the interaction layer is not designed appropriately and conversely, interaction at the ''interaction'' and ''semantic'' level can be achieved only if the other layers are sophisticated enough.

V. REPRESENTATION LAYER

In this layer we distinguish five different parts: representation of (i) the domain, (ii) itself, (iii) the context, (iv) other smart objects, (v) the user, and (vi) domain and task specific knowledge. Notice that (iii), (iv) and (v) are kept separate to emphasize three areas of adaptation for an ARSPO.

A. CONCEPTUAL MODELLING

A first general decision to be adopted by the designer is the level of conceptual representation of the features and properties of the concepts that are used for modeling itself, the user, the context, other objects and task. A range of alternatives is available: at one extreme, the designer can adopt an ontology for modeling these concepts and the relations among them; at the other extreme, all the concepts can be simply represented as atomic labels without any further specification. The ontology may concern any of the categories of knowledge (structural, functional, behavioural, teleological) and can be designed at different levels of sophistication, including coarse or detailed descriptions of the concepts and their properties. In particular, different levels of details may be adopted for different types of knowledge (e.g., fine grained description of the concepts concerning the goals of the user and coarse grained descriptions for those concerning the structure of the context).

As an example regarding the smart chair, the ontology may model concepts such as the materials (wood, metal, leather) and their properties or concepts regarding the user such as ''back'', ''backbone'', ''working'', ''deadline''.

B. MODELING ITSELF

First of all, the designer should decide if and how the object should be aware of itself. This involves analyzing requirements on the type of knowledge about itself the ARSPO should be able to manage, concerning both its physical and digital counterparts. For each one of them, an object may be aware of its structural or functional or behavioural or teleological dimensions. The concepts used in this representation may refer to a domain ontology and thus their properties may also be related to those used in other parts of the model (e.g., context or other ARSPOs).

For example, consider the running example of the smart chair: as regards the physical aspect, the chair might know about its material (structural dimension) or about its function (offering people a seat); as regards the digital dimension, the chair might know where sensors are embedded in it (structural dimension) or that it can offer the user the opportunity to measure her blood pressure (functional dimension) or that its goal is to support people in monitoring their health state (teleological dimension).

C. MODELING THE CONTEXT

By context we mean the environment surrounding the ARSPO, excluding other smart objects and people. As in the case above, the designer may refer or not to concepts in an ontology and the four dimensions above should be taken into account concerning either the physical or the digital context (or both). Structural properties of the environment might include information on the objects that are present and their physical/digital properties, while teleological properties might describe the purposes of these objects. Also in this case we may have a flat representation or a more structured one including also mapping between context features.

Let us consider again the example of the chair. In order to provide adaptive recommendations it could be relevant to take into account information such as the time of the day (structural information) or the presence of a wi-fi (digital structural information) or about the purpose of the environment (e.g., a studying room vs an office vs a dining room at home with its corresponding functions and goals).

D. MODELING OTHER SMART OBJECTS

Other smart objects may be present in the environment where the ARSPO operates and the ARSPO may interact with them in performing its tasks. Thus these SPOs may need to be modeled by the ARSPO. In particular, this means to model those pieces of information that might be relevant or useful for the ARSPO in order to carry on its tasks. This includes the problem of defining communication protocols for guaranteeing correct and fruitful interaction and data exchange among objects. As in the case above, this may concern digital or physical properties (or both), may include ontological representations of the concepts involved and may involve one or more of the four dimensions.

A further aspect that is relevant in the case of smart objects is the fact that an ARSPO may maintain ''social relations'' with other smart objects and thus have a model of these relations. Examples of possible social relations among objects (based on Fiske's theory [20] commonly used in sociology) could be the following ones:

- ''Parental object relationship'': among objects belonging to the same production batch, i.e., usually homogeneous objects originated from the same manufacturer.
- ''Co-location object relationship'': among objects (either homogeneous or heterogeneous) used in the same place, e.g., sensors, actuators, objects in a smart home).
- ''Co-work object relationship'': whenever objects collaborate to provide a common application (e.g., emergency response, telemedicine, etc.).
- ''Ownership object relationship'': among heterogeneous objects which belong to the same user (mobile phones, music players, game consoles, etc.) .
- ''Social object relationship'': when objects come into contact, sporadically or continuously, because their owners come in touch with each other (e.g., devices belonging to friends or colleagues).

Let us consider again the example of the chair. In case the context includes other SPOs in the room such as an intelligent lighting system or an intelligent fridge. For example it may have a ''co-work'' relations with the lighting systems (suggesting how to tune the lights when the user is sit at the computer) or a ''co-location'' relation with the fridge for tuning the temperature of refreshments during work in hot days. Moreover it could maintain an ''ownership'' relation with the user's smartphone, for example, blocking certain types of calls when the user is at work.

E. MODELING THE USER

Modelling the user is a fundamental aspect of adaptive systems [6], [7], [19]. The user can be considered as part of the context, but we prefer to keep it separate in order to focus on her and better consider her features, as in traditional user modeling and adaptive system community [30]. Also in this case the aspects discussed above should be taken into account. The model may concern any of the four dimensions of representation:

- structural knowledge, e.g., user's physical features
- functional knowledge, e.g., user's preferences
- behavioural knowledge, e.g., user's habits
- teleological knowledge, e.g., user's goals and plans

When defining the user modeling requirements for an ARSPO, the designer should keep in mind all the factors that could be relevant for selecting the information or service to be provided (content adaptation) and the way they are provided (interaction adaptation). The representation may be flat (simple attribute-value pairs [15], probability distributions [9], fuzzy intervals [10], plain vectors [51] and bags of words [12], vectors associated with weights, such as Vector Space Models [17], [40]) or may include mappings for relating users' features among themselves, possibly taking into account also contextual features. In this case, a conceptualization with users' features may be adopted (e.g., an ontology of users' features [27], [33], [35], [45]) and the concepts in the user model may refer to a domain ontology. This is the case of overlay models, where domain-dependent user features, such as interests or knowledge, are represented as an overlay of the domain structure. For each item in the domain, the user's current state (e.g., interest or knowledge) with respect to the item is recorded [8].

Social relations of the users with other users may be modeled (e.g., parental, friendship, colleagueship, ...). The ARSPO should mantain also its relation with the user (e.g., as ''ownership'' (the user who owns the ARSPO), ''used by'' (the users who use the ARSPO), ''friend'' (relation between the ARSPO and a user). These relations may be exploited by the ARSPO in the recommendation process.

Let us consider the example of the smart chair for a smart office with office employees as target users. The chair may model structural information such as the user's height and weight as well as behavioral information such as the user's preferences (e.g., about the seatback inclination) or goals (e.g., finishing some work before a close deadline) or it can maintain information on which other objects are owned by each user.

F. MODELING DOMAIN AND TASK SPECIFIC KNOWLEDGE

Domain modeling requires a more articulated discussion since it involves representing different concepts, such as those related to the services that can be provided, the interaction modes that can be used and the mappings for selecting them, given user, object, context features.

1) DIMENSIONS FOR THE SERVICE TO BE PROVIDED

Several aspects have to be considered as regards the representation of the services that can be offered:

- First of all, the level of representation of the services which may be simple labels or can refer to concepts in the domain ontology.

- Second, the nature of the services should be defined. The nature may range from purely digital (e.g., information to be provided to the user) to purely physical (e.g., an audio/tactile stimulus, possibly a feedback to some user's gesture), with a spectrum of intermediate mixes.
- A further aspect is the static/dynamic nature of the services, that is: can the set of services be defined statically or do they change dynamically and have to be searched for by the object at run-time (e.g., on the Internet)?
- A further aspect is the complexity of the services, that is: are the services simple or should they be assembled by the object by composing elements (from its repertoire or looked up in the Internet).
- The type of representation, i.e., whether the structural, behavioural, functional, teleological dimensions of the services are modeled.

2) DIMENSIONS FOR THE INTERACTION MODES

An ARSPO should maintain a model of the interaction modes it can offer to users. Being a physical object makes a fundamental difference in this modeling process since the interaction modes offered by an ARSPO mirror the dual physical and digital nature of the object. A number of aspects have to be considered:

- First, as in the case above, the level of representation of the modes can vary from simple labels to the adoption of ontologies. In particular, in this case the ontology can model the physical and/or digital nature of the modes offered to the user (possibly with links to concept modeling the user, context,..).
- A second aspect regards the integration of the physical digital/dimension, ranging from purely physical to purely digital to integrated modes where the two dimensions are mixed in a seamless way.
- As in the case above, the modes may be statically defined or they can vary.
- The fact whether the modes involve other objects (or smart objects) or not and in particular whether they need collaboration with other SPOs.
- The type of representation, i.e., whether the structural, behavioural, functional, teleological dimensions of the modes are modeled.

3) ADAPTIVE RECOMMENDATION MAPPINGS

Formula (1) defined in Section II is used here to represent adaptive recommendation as a relation:

$$itself \times cxt \times objs \times user \rightarrow Service \times InteractionMode \tag{2}$$

Where:

- $itself$, $user$, cxt and $objs$ are the sets of the features concerning the ARSPO itself, the user, the context and other smart objects that are relevant for adaptive recommendation;
- $Service = \{S_1, S_2, \ldots, S_n\}$ is the set of services that can be provided by the ARSPO;
- $InteractionMode = \{IM_1, IM_2, \ldots, IM_m\}$ is the set of interaction modes that can be used by the ARSPO;

and are defined as discussed in the subsections above.

An ARSPO may maintain a number of mappings with the form above that can be used to select the appropriate service and interaction mode. The mappings, in particular, can be read as follows. Each mapping relates the mentioned features (features of the ARSPO, the context, the user, social relations, a combination of them) to the service to be provided and the appropriate interaction mode. We do not take into account how this knowledge is actually represented since we are only interested in the conceptual mapping. Other mappings may define relations among the sets *itself*, *user*, *cxt* and *objs*; for example, associations between user and context features.

In conclusion, let us consider again the chair. The services it can offer may have a mixed physical-digital nature; the interaction modes may be physical (stimulus to the user) or digital (turning some alarm on). The recommendation mapping may be very simple (e.g., rules mapping the user position to alerts) or more complex (by taking into account the user's goals, contextual features, the fact that the user is working and mapping them to alerts and to messages for coordinating other SPOs). For example, tuning an optimal combination of inclination, firmness and heating.

G. DISCUSSION ON DESIGN CHOICES

In the subsections above we identified different types of knowledge that can be modeled in an ARSPO. In this section we analyze these alternatives and we discuss how they impact the level of smartness and the complexity of the ARSPO being designed. By smartness here we intend the level of consciousness that the ARSPO has about the world surrounding it; the analysis can apply independently to the types of knowledge in the previous subsections, leading thus to a multi-faceted characterization and classification of ARSPOs as regards their models of the world. As we shall discuss later in more detail these levels correspond to different levels of complexity in creating and managing the models.

A first dimension for analysing the level of smartness is the availability of a conceptual model of the world of application. At one extreme we may have an ARSPO with a sophisticated domain ontology, modeling all features of the domain such as user and context features, service and interaction mode properties. At the other extreme we may have an ARSPO with no model at all and for which each feature is simply a label with no associated meaning.

Moving to the types of knowledge that are represented in the model, the different categories of knowledge we introduced correspond to different levels of awareness an ARSPO can have about itself, the user, the context, other objects, the world around it. The sequence teleological-functional-behavioural-structural corresponds to a decreasing scale of sophistication on an ARSPO knowledge base. From the ARSPO point of view this corresponds to a decreasing

TABLE 2. Types of knowledge in a smart chair.

Category	ontology	itself	cxt	user	objs	domain (i) services (ii) modes (iii) mappings
Structural knowledge	Ontology describing the parts of the chair and their properties and relations.	- The fact that it is made of wood. - Its sensors (e.g., pressure) and actuators (e.g., vibration).	Objects in the environment.	User's weight or height	Interfaces offered by the PC	(i) The variation of the seatback that can be offered; (ii) The feedback provided to the user back (iii) Mapping between user's height and seatback inclination
Teleological knowledge	Ontology of chair's or users's goals;	Minimizing energy consumption.	Being unobtrusive wrt other objects ensuring a comfortable sitting and a safe position to user.	User's goals: finishing some work, preserving her well being	Goals of the PC: providing work support, providing entertainment	(i) The goals of the services it can offer, (ii) the goals of interaction modes (iii) Mapping user's goals to the chair goals
Functional knowledge	Ontology of functions of the chair;	Gathering data about user behaviour (posture, ...).	Driving other objects to stimulate a user reaction (turn on the light, ...).	Tasks that the user can perform, such as carrying on a specific work	Functions offered by the PC, e.g., command to play music	(i) The fact that the setback can move; (ii) The fact that the user can press the setback with her back; (iii) Functions that can be offered in response to the function performed by the suer
Behavioural knowledge	An ontology describing the types of movements of the chair;	The way sensor data are collected; reactions to sensor data.	The way to interact with other objects.	The way the user behaves when carrying on a specific work	The way the PC behaves when the command to play music is activated	(i) The ways the setback can move; (ii) The ways the user can press the seat=back (iii) Mapping between user's behaviour and the way the chair reacts

awareness about the concepts being represented (for example when referring to knowledge about the user this corresponds to a decreasing awareness in the user model). From the point of view of a user the sequence corresponds to a decreasing level of perceived intelligence of the ARSPO. Behavioural knowledge without functional one corresponds to knowing how to act without knowing what one can actually do: we can thus argue that an ARSPO having only behavioural knowledge is less aware (and less smart) than one having functional knowledge. It must be noticed, however, that having functional knowledge without behavioural one may correspond to create a potential without the ability to actually perform. Similar considerations can be made for comparing the functional with the teleological level. Teleological knowledge corresponds to the highest level of consciousness, but at the same time teleology without functional can result in purely speculative abilities. These considerations hold both for knowledge about itself (resulting in different degrees of self consciousness) and for knowledge about the world itself, context, other SPOs, users).

As regards social relations, also in this case we can isolate a spectrum of levels of consciousness for each one of the two cases: relations with people and relations with other SPOs. At one extreme we have an ARSPO that does not maintain any relation with either people or other objects. At the other extreme we have an ARSPO that maintains all types of complex relations discussed in the previous subsections. All intermediate cases define also in this case a lattice of levels for comparing ARSPOs.

Let us now analyze domain knowledge and, in particular adaptation mappings. The level of sophistication of the ARSPO knowledge base depends on many factors in this case. First, it depends on the complexity of the set of mappings on how the set covers all the situations in which the ARSPO can operate (i.e., how much it covers the combinations of concepts that can appear in user, models, in the context). Second the sophistication depends on the type of knowledge used in the mappings. For example, mappings in which the premises involve teleological knowledge about the user can be regarded as more sophisticated than mappings based only on structural knowledge about the user. Similar considerations hold for the other types of knowledge in the premises of the mappings leading to a complete lattice for comparing them. Finally, the level of sophistication depends on how the sets *Services* and *InteractionModes* are defined. In the simplest case they can be characterized as static (pre-defined) sets. Sophistication can be added if the sets are not static and predefined (and thus the ARSPO can assemble the service dynamically) or in case the elements have to be generated by composing simpler services (predefined or located dynamically).

Let us consider the example of the smart chair. We assume that the chair can monitor the environment (collecting data such as temperature, brightness) and the user (collecting physiological data, e.g., her heart rate and blood pressure, or postural data). The chair has actuators for modifying its shape and the seat, it can activate heating and modify its height. The chair can also provide feedback to the user concerning her state and position. Finally, we assume that it can communicate with other objects (e.g., with the PC on the user's desk or the lights in the room or with the table). Table 2 contains examples of types of knowledge that can be modeled for this ARSPO.

VI. REASONING LAYER

This section (layer) and the following two concern the ability to make inferences on knowledge and data gathered from the interaction with the world. However, we distinguish three layers, separating three aspects that are relevant in the design of an ARSPO. In this section we discuss the generic ability to make inferences on data using knowledge. In the next section we focus on the specific task of an ARSPO: adaptive recommendation and we discuss the reasoning techniques that can contribute to this type of problem solving. Finally in the following section we analyze learning, which exploits inferences for a completely different conceptual task.

Inferences in an ARSPO may be activated in different situations. They may be started by some external perception (physical or digital), corresponding to data gathered by the ARSPO's sensing system. They may be also started autonomously by the ARSPO itself in certain specified conditions; finally it could be activated on request from the user or other SPOs. Inferences involve data perceived by the ARSPO (at physical and/or digital level) and may involve the types of knowledge discussed in the previous section.

- **Inferences concerning knowledge about itself.** An ARSPO may infer information about itself from sensory data or from the interaction with people and/or other ARSPOs. This may range from simple inferences for determining the behaviors/functions that are activated/requested to complex inferences for hypothesizing functions or goals to be achieved. These inferences can be deductive, exploiting knowledge about itself (mappings between user features, as mentioned in the previous section) or abductive, whenever they require making hypotheses from sensor data. It may also be based on analogy principles, exploiting forms of case-based reasoning reporting a situation to previously encountered ones.
  As an example, the chair may infer (or hypothesize) the status of its surface from information about the temperature (the surface is hot if the weather is sunny and temperature very high).
- **Inferences concerning knowledge about the context.** Similarly an ARSPO may exploit knowledge to expand information about the current context of operations. Also in this case input to the process may come from the sensor systems or possibly from the user or other SPOs. The forms of inference involved are the same as above. For example, the chair may infer (deduce or hypothesize) that the temperature is high from the fact that air conditioning is turned on.
- **Inferences concerning knowledge about other smart objects.** This case is in part similar to the previous one. In addition inferences may be related to communications coming from other objects. The chair may know about the physical and digital properties of such objects (e.g., the functions that they can offer, at physical and digital level); it may then be able to reason about the tasks that the chair can ask them (e.g., switching some lights when the user is sit for working) and how to interact and maintain relations with them.
  An ARSPO can maintain a graph of social relations with other objects and with people. The ARSPO should be able to perform some forms of "social activities" with respect to other objects, for example:
  - interact and communicate with other objects in order to know about their functional, structural, behavioural and teleological knowledge and to cooperate and negotiate with them;
  - manipulate information from social networks and share information with others members;
  - identify new interesting peers and establish a connection with them.

  The establishment and management of such relationships should be in charge of the objects, following the rules on the objects' social interactions established by designers.
- **Inferences concerning knowledge about the user.** This is the ability to manage the user model, i.e., the ability to infer information on the user from other pieces of information (e.g., from the context). The requirements should describe which sorts of inferences are needed, given the pieces of information that can be gathered directly. Also in this case the forms of inferences involved are deduction (exploiting knowledge about users to infer features from observed one), abduction (explaining observations by hypothesizing user's features), analogy (inferring user's features from features of other similar users). They may range from inferring physical information about the user to infer her goals.
  Let us consider the example of the smart chair for a smart office with office employees as target users. The chair may infer user's goals explaining abductively her actions or may hypothesize user's preferences for the seatback from her height and from information about the temperature and time of the day.
- **Inferences concerning knowledge about the domain.** In this case the ARSPO could be able to make inferences on the services and interaction modes, starting from knowledge about them. For example, it should be able to infer which services (modes) are actually available or not before starting the adaptive recommendation process. The forms of inference that can be involved are similar to those mentioned above.

A. DISCUSSION ON DESIGN CHOICES

As in the case of the representation layer we can single out different levels of sophistication of ARSPO reasoning abilities. In particular, two dimensions can be isolated to this purpose. The first dimension concerns how many of the types of inferences mentioned above are considered. An ARSPO, in fact, can implement only some of them, depending also on the availability of knowledge supporting the inferences. A second dimension is related to the form of inferences that the ARSPO can perform. In particular, for each one of the types of inferences above in fact, the ARSPO may implement one or more inference patterns. For example, it may only be perform deduction starting from collected data using mappings in its knowledge base, or it may implement also forms of abduction for explaining observed data or even it may also be able to perform forms of analogical reasoning or forms of non-monotonic inference (e.g., drawing conclusions by default or using close world assumption). The ability of adopting more than one pattern increases the levels of sophistication of the ARSPO and thus the level of intelligence that the user can perceive. We can also argue that implementing forms of common-sense reasoning, e.g., qualitative form of default reasoning or analogical reasoning gives the user the sensation of an object with ''more human'' inference capabilities.

VII. PROBLEM SOLVING LAYER

In this layer we consider the design decisions concerning the specific problem solving activity of adaptive context-aware recommendation performed by a smart object.

Problem solving in an ARSPO can be characterized as the problem of making a number of decisions which concern the service to be offered (WHAT), the interaction mode to be adopted (HOW), the situation in which this process takes place, if any (WHY and WHEN), given the specific user (WHO) and context (WHERE). Differently from a software agent, an ARSPO can perform these choices at several levels, involving its physical and/or digital dimension. It can change its structure or modify its affordances; it can adapt its function or behavior and thus the services it offers. The highest form of intelligent adaptation corresponds to adapting its own goals to the goals of the user.

Interaction is important in adaptive systems, both as a way of getting information about the user and context and as a way of putting adaptive behavior into practice. The physical nature of an ARSPO has a peculiar role in this interaction as users expect natural forms of interaction. Knowing itself, its user and the context, the ARSPO can select a proper *interface* to provide recommendations. Intelligent forms of adaptation require sophisticated forms of interaction (from syntactic to semantic and interactional).

For the sake of simplicity let us decompose the task into subtasks: (i) recognizing the user's peculiarities, (ii) selecting the items to be recommended and (iii) presenting them in an adapted way. The first task can range from a simple form of *data collection* to complex forms of *abductive reasoning* to analyze the user's behavior or to interpret her needs and goals. The second and third tasks can be performed in different ways. A simple way is to reason *deductively* relating user preferences to the features to be tailored and to the most appropriate presentation. Adaptation can be also implemented as a form of *analogical reasoning*, by providing the user with something that other similar users appreciated in previous interactions.

In some cases an ARSPO may need to cooperate with other objects. Thus it can exploit the social relations with other SPOs and exchange information with them. At the simplest level, the interactions can be simple and isolated (e.g., requests for performing a task). At the highest level an SPO may be able to participate (or coordinate) forms of cooperative *planning* or *choreographies* in order to provide the best service to the user.

The exploitation of social relations with other objects may be relevant in this case as the ARSPO can select the SPOs that are most suitable for collaboration, given the specific situation. Simple examples are exploiting co-location to select SPOs that are in the same place, or ownership relation to select SPOs that belong to the user. Advanced examples might exploit for instance game theoretic analyses to support decisions when the actions of the SPOs are interdependent. This approach has been already applied in multi-agent systems (for example [43]).

Notice that in the paper we assumed that recommendations have to be provided to a single user and not to groups. Providing recommendation to groups involves the ability to reason on multiple user models and on social relations about the people in the group to make the most appropriate suggestion [16], [50]

Let us consider the example of the chair. It may adopt different approaches for sensing the user (e.g., textile material between the user and the chair seat or seatback, proximity sensors detecting the user presence, etc.). It can choose the type of recommendation (e.g., physical, by changing its configuration, or digital, by sending messages to the user) and different ways to communicate (e.g., actuators providing physical feedback or messages to the user's smartphone). Finally the chair may exchange information or cooperate with other objects such as the user's computer, her desk or the lighting system.

A. DISCUSSION ON DESIGN CHOICES

In summary, we can single out a number of design alternatives in the problem solving layer, leading to different levels of sophistication of the resulting ARSPO:

- The adaptive recommendation may have only physical nature or only digital or can be a blending of the two.
- The level at which service adaptation is performed, ranging from physical adaptation wrt physical user's feature to the extreme of goal adaptation wrt user's goals.
- The level at which interaction adaptation is performed, ranging from syntactic to interaction levels.
- The level of sophistication of the reasoning process in the tasks above; in particular at one extreme the ARSPO may perform simple lookup of the solution that best fits the available input data. At the other extreme it may perform complex forms of common sense reasoning, possibly combined with other inference patterns. The number and complexity of reasoning patterns mirrors the sophistication of the process and leads to different levels of proactiveness perceived by the user.
- The level of collaboration with other SPOs, taking into account social relations with and among them.

Thus, also in this case, we can single out a spectrum of alternatives leading to different levels of smartness (in term of problem solving sophistication) and consequently different levels of complexity (in terms of complexity of implementing the reasoning strategies and computational complexity of the reasoning process). Static services and modes and no collaboration with other SPOs leads to a simple ARSPO with limited level of smartness but simple to implement and fast to provide recommendations. Conversely the ability to assemble the service dynamically, by exploiting data gathered from the user and context and possibly cooperating with other SPOs leads to an ARSPO with a higher level of smartness but it may be complex to implement and may require powerful computational resources that might not be available in small constrained devices.

VIII. LEARNING LAYER

Learning is particularly important in adaptive recommenders and in the way the recommendation process is perceived by the user. Learning is the ability of an ARSPO to ''meta-reason'' on its problem solving activity with the aim of improving its knowledge and abilities, based on the analysis of previous experience and user feedback. As explained in Table 1, learning is a reasoning activity which typically results in revising/updating/refining/extending the ARSPO knowledge base. Thus, it requires specific design decisions on how, to what extent and how fast the ARSPO knowledge has to be updated. Given the relevance of such design decisions and the impact on the other layers, we kept it separate from the reasoning layer.

With reference to Table 1, learning can be analyzed by taking into account different dimensions.

(i) Input data. They come from the ARSPO ability to observe its own problem solving activity. Potentially all the sensed data can be made available to the learning process and they can be augmented by data coming from introspection to define the context of a problem solving event. The latter includes the inferences made by the reasoning layer and, more interestingly, the decision made by the ARSPO as regards the different choices of adaptive recommendation (''what,'' ''when,'' ...).

(ii) Target. The learning process can have different targets, in terms of types of knowledge (learning about itself, the context, the user, the domain, other objects), and different dimensions inside each type of knowledge (structural, behavioural, functional, teleological). In principle, learning is not be limited to the representation layer but it could also impact other layers. An ARSPO adopting a combination of reasoning strategies could learn that one of them is not appropriate for certain types of users and decide to change its strategy. For example, an ARSPO adopting analogical reasoning to make recommendations to a user based on the choices of other similar users (i.e., collaborative filtering) may decide to disregard this approach in favor of strategies that relate user preferences to content features adopting deductive inference strategies.

(iii) Feedback. Different types of feedback (from the users, the context and other smart objects) may be available to the ARSPO. We can distinguish between supervised learning (trained with classification examples, i.e., positive and negative feedback from the user about the content of the recommendation or the time when it has been delivered), unsupervised learning and reinforcement learning (using a trial-and-error approach). Feedback may be implicit (how the user interacts with the object) or explicit (asking for an explicit feedback, e.g., on a binary or a graduated scale).

(iv) Methods and algorithms. Different algorithms can be used, depending also on the three dimensions above (for example decision trees, clustering, Bayesian approaches, rule-based approaches, neural networks, etc.).

(v) Time. Learning can take place at different time: it can be activated periodically (e.g., with fixed timing or depending on the usage) or it may be event or trigger-based (e.g., whenever the number of negative feedbacks is over a threshold). The revision may be performed while the system is running or at given times.

Combinations along these dimensions lead to different learning capabilities.

Let us consider again the example of the smart chair. The chair may learn from user's feedback that she does not like to receive alerts when she is working with a very close deadline or that typically the user has a break every day at the same time and thus can avoid sending alerts whenever the break time is approaching. In the former case it uses negative feedback from the user; in the latter learning is not supervised.

A. DISCUSSION ON DESIGN CHOICES

As in the other layers, also in the learning one we can single out different choices leading to different levels of intelligence for learning abilities. At the lowest level there is the case where no learning occurs. At the highest level we may have an object which can learn all types of knowledge and possibly also its reasoning and recommendation strategies from the interaction with the user (interpreting users' feedback without asking explicitly extra feedback). Timing is more controversial, since in some cases frequent refinement of rules may be preferable while in other cases it may lead to behaviors of the ARSPO that may be perceived by the user as contrasting, unpredictable and annoying. An advanced strategy would be tuning learning time depending on the effects of the learning process on the user. Intermediate cases lead, like for the other layers, to a lattice of design alternatives.

IX. CONCEPTUAL MODEL FOR ARSPO DESIGN

In the previous sections we analyzed the layers of our model, we singled out the choices that a designer can make at each layer, discussing the consequences of such choices in terms of levels of smartness and complexity of the resulting ARSPO. In this section we provide a guideline to design an ARSPO. The same steps can be used as a checklist to compare ARSPOs according to the levels of intelligence they embed and exhibit to users.

Let us start from one example in order to clarify the presentation, considering again the running case of the smart chair.

- In the **preliminary phase** the designer selects the specific chair to be augmented, focusing on workers as target users and on the purpose of improving their posture when they sit at their work desk.
- In the **interaction layer** the designer makes decisions about the interaction between the ARSPO and the user. Preliminary to this is the decision on the integration between the digital and physical parts of the object. The designer must decide how to add digital sensors and actuators to the chair and the type of interaction they have to support. If the designer can participate to the design of the chair, then she can integrate more easily the sensors/actuators with the physical components of the chair. If not, she can simply over impose some digital parts and provide a dedicated interface (GUI-like or TUI) to the user or she can work towards seamless interaction. Finding the appropriate affordances is the key issue. The designer may discover, for instance, that users tend to lean toward the seatback when they relax or are tired and thus she can use this knowledge to build appropriate affordances and adaptation/recommendation strategies. In this way the designer can range from simple interaction at task level (touching the GUI to get feedback about the position) to interaction at semantic or interaction level with seamless user support.
- In the **representation layer** different choices have to be made according to the different parts of the ARSPO knowledge base. First of all, the designer may choose whether to adopt a domain ontology or not and if she decides to adopt it, which parts of the domain have to be modeled (e.g., physical features of the chair, properties of the user, properties and relations among the objects in the context,). The designer may choose to have a sophisticated user model involving a representation of the user's goals but a weak representation of the context and no representation at all of other objects in the environment. However, she could also choose to have a weaker user model involving only information about user's structure but a sophisticated model of the context involving descriptions of the functions of all the objects in the environment. As regards the domain model, the designer can chose to have weak descriptions of the services to be provided or more sophisticated descriptions involving the effect of each position of the seatback. Similarly the designer can choose to have a weak representation of the interaction modes involving only their identification or a more sophisticated model involving the representation of the type of interaction they support. The model can be made deeper and deeper by representing the properties of the pieces of information and services, resulting in an ontology of the domain. Similarly, adaptation mappings can range from very simple ones (e.g., using the user's structural features such as her height and weight to select the most appropriate sitting position), to more complex ones that take into account the user's goals (e.g., finishing a paper on the computer), contextual information, the status of other smart objects or even can request data to other smart objects (e.g., from the fitness tracker of the user) to provide the best support to the user (e.g., recommend some behaviour and select the best mode to warn the user about the best sitting position). Choosing the complexity of recommendation mappings is partially dependent on the choices above, in the sense that, for example, they can provide sophisticated mappings between user's back and action on seatback goals only if all these goals are represented.
- In the **reasoning layer** the designer can choose to have different levels of sophistication in all the reasoning phases isolated in Section VI. This is strictly dependent on the type of knowledge modeled in the various parts of the representation layer. For example, the designer may choose to have complex inferences for deriving information about the user from sensor data (e.g., inferring her goals from her position, the time of the day and information from her computer) or, conversely, the designer may choose to avoid these inferences and use sensor data only. Similar considerations may apply also to the other types of inferences.
- In the **problem solving layer** the designer may choose different strategies for making recommendations to the user. These decisions are dependent on many of the choices made above, in particular in the interaction and representation layer. The designer may consider a very simple strategy, such as looking for choices previously made for the same or similar users, to complex strategies involving the use of knowledge that maps user to service preferences. For example, the chair may perform analogical reasoning to decide which settings are suitable for a user, given her weight and height and given the choices for similar users, or may adopt more sophisticated forms of inference to determine the best setting, given the user features, the settings it can offer and mappings relating users features, context situations and settings.
- Finally, in the **learning layer** the designer may choose how and when the chair should learn from previous experience and possibly user feedback. The simplest option is to disregard learning, but in this case the chair will keep iterating the same behavior also if the user provided negative feedback. On the other hand, the designer can decide to adopt a simple learning strategy, such as caching the recommendations for which the user provided positive feedback for re-proposing them in the future to the same or similar users or more complex strategies involving the collection of all available data (e.g., data from sensors, contextual conditions and user's feedback) trying to infer general rules from them (e.g., using supervised induction strategies).

The example above shows that many alternatives are available to a designer and that she can make many different choices as regards the level of sophistication of the resulting ARSPO, with many dependencies among the choices. Generalizing from the example we can single out a lattice with different dimensions for comparing ARSPOs as regards their level of smartness. The dimensions of choice have been summarized in Figure 2. The designer can follow the steps, analyzing each dimension, starting from the requirements for the object being designed. The same steps can be used as a checklist for analyzing an ARSPO or for comparing two ARSPOs.

There are cases where the choices can be made independently on each step; in other cases some dependencies have to be taken into account. For example, sophistication in the reasoning strategies may be useless if the knowledge is too shallow. On the other hand, the level of sophistication in user modeling can be chosen independently of the level of complexity in context modeling and in learning strategies.

In conclusion, one important consideration is the relation between the level of sophistication of the design choices and the complexity of the ARSPO to be implemented. Adding sophistication and thus increasing the level of intelligence (and of intelligence perceived by the users) adds complexity to the implementation of an ARSPO. Complexity here means at least two aspects: on the one hand, the design becomes more complex, in terms of knowledge, inference strategies, learning strategies, .., to be implemented (depending on the choices made in the lattice of alternatives). On the other hand, the recommendation process to be carried on by the ARSPO can become computationally more complex, imposing requirements on the computational power to be embedded on board the ARSPO or made accessible to it.

Let us consider some simple examples. Adding an ontology to represent and relate the concepts used to model the user, the context or the domain can significantly improve the level of intelligence of an ARSPO that, in this way, can exploit the meaning of the concepts in order to provide better recommendations in a more effective way. On the other hand, when existing ontologies are not adequate to the designer's needs, the ontology has to be built and this is not easy in many cases, especially if detailed descriptions and complex relations among concepts have to be represented. Moreover, reasoning with an ontology can be computationally expensive requiring access to significant computational power. Similar considerations apply also to the other choices discussed in the paper. As a further example, learning is a crucial feature for being perceived as an intelligent entity. However, learning requires managing and maintaining information on the decisions made and on users' feedback, the ability to interpret these feedback and the implementation of algorithms that extract information for updating/revising/creating the knowledge base starting from this analysis. Also in this case both design and computational complexity is increased (although in this case most of them can be off line and do not need to be embedded in the ARSPO itself).

X. CASE STUDIES

In this section we complement the description of the model with an exemplary analysis of its application to existent ARSPOs. We apply the model to a research prototype that we developed (PosturalTS) and to a commercial system (Jawbone UP). The analysis is useful to show that the model can be used both to support ARSPO design and classification.

Project name: PosturalTS

Partners: University of Torino, Departments of Computer Science, Biology, Sport Sciences, 2016-2017

Description: A t-shirt for monitoring people's position at work, providing physical feedback to a user when she is in a bad position or when she stays in the same position too long.

1) interaction layer:
   - level of integration: TUI (sensors and actuators embedded in the t-shirt)
   - affordances: physical affordances (notification via actuators in the t-shirt)
   - interaction level
2) representation layer:
   - knowledge about domain: no
   - knowledge about itself: structural knowledge concerning the position of the sensors
   - knowledge about context: no
   - knowledge about user: structural and behavioural knowledge, no teleological knowledge
   - knowledge about objects: no
   - knowledge about tasks: functional and behavioral description of the services, rules for the mapping among user physical and behavioral data
3) reasoning layer: inference for extending knowledge about users, given information from sensors
4) problem solving layer:
   - service selection: deductive reasoning on data in order to correlate data gathered and inferred from sensors to user's behavior and thus to alarms ands signal to be provided (structural and behavioral knowledge)
   - mode selection: dynamic, simple, physical
5) learning layer:
   - input data: data from sensors and inferred by the systems
   - target of learning: user behavior (learning user habits in order to support users with suitable alarms)
   - approach: unsupervised learning; when: after repeated feedback, revise user model

System name: Jawbone UP[1]

Developer: Jawbone Company, 2014

Description: a soft rubber wristband that resembles a mini coil, with the main goal of tracking user sleeping and walking habits, as well as provide haptic alarm and digital visualisation of the traced data.

[1] https://jawbone.com

1) interaction layer:
   - level of integration: TUI and GUI
   - affordances: both physical and digital affordances
   - task level
2) representation layer:
   - knowledge about domain: no
   - knowledge about itself: functional and behavioural knowledge
   - knowledge about context: only structural knowledge
   - knowledge about user: functional and behavioural knowledge, teleological knowledge (flat user model)
   - knowledge about objects: no
   - knowledge about task: no model of services, no model of modes, rules for the mapping among user goals (teleological) and service (alarm)
3) reasoning layer: no inference
4) problem solving layer:
   - service selection: deductive reasoning on user's goals in order to activate some functionalities (such as vibration to wake up the user): dynamic, simple, physical;
   - mode selection: dynamic, simple, physical (alert) and static, simple, digital (visualisation)
5) learning layer: no

XI. RELATED WORK

ARSPOs are growing fast in different fields. Examples of products on the market are wealth and fitness devices that provide users with recommendations about training and well-being practices, smart home assistants, help tools for emergency management, etc. In research projects, ARSPOs have been developed in the context of augmented reality, robotics and the IoT environment. For example, [21] describes a distributed multi-agent recommendation system that is designed to suggest resources for applications such as urban computing, smart cities and health care. The agents manage the thing descriptors and exchange them on the basis of ad-hoc probability functions. Another example is described in [29]. The context-aware recommendation system suggests a set of micro-services used to orchestrate networks of smart objects taking into account users' needs and preferences in smart spaces.

In the literature, several efforts have been made to develop classification and design models for SPOs, while no guidelines have been provided to design adaptive smart physical objects. The classification models can be divided in two main approaches: non-nested and nested approaches [28]. Non-nested approaches present features which do not reflect increased complexities; while nested approaches propose a stratification of features from simpler to more complex ones. Our classification is closer to the latter even though some similarities with the features that describe the objects can be found also in some non-nested approaches.

Among the *Non-nested approaches*, one of the classifications with some similarities with our layered model is [36]. In their model, objects can combine the following characteristics: (I)dentity, (S)ensing, (A)ctuation, (D)ecision-making and (N)etworking. I stands for identity and the storage of any other relevant data, S stands for sensing its physical condition and the environment, A for actuation of internal or external devices, D for decision making and participation in controlling other devices or systems, N for networking to reach and receive information through a (wired or wireless) network. Such characteristics can be mapped to our layers in the following way: (I)dentity happens in the interaction and representation layer, (S)ensing and (A)ctuation in the interaction layer, (D)ecision-making in reasoning and problem solving layer and (N)etworking, in reasoning, problem solving and interaction layers. Even though each feature can be mapped to different layers of our model, our classification approach allows to compare objects not only for their capabilities but also for the level of such capability and for the interaction with the other capabiliteis. In this respect, our approach is more similar to nested-approaches.

Among *Nested approaches*, [38] proposes a model which distinguishes three levels of intelligence: information handling, problem notification and decision making. The authors consider smart products as a particular class of smart objects, focusing mainly on the internal behavour of the objects and less on the interaction with the users.

Similarly, [28] presents a model for classifying smart objects using capabilities; the model can distinguish objects with basic capabilities from those with complex ones. They characterize core and optional capabilities. The former are atomic and simple (digital identification, retention, communication, energy-harvesting); the latter are organized in four dimensions according to what they address: internal factors, environment factors, human factors, and engineering factors. The mapping with our model is not simple, since all the capabilities can be split on different layers.

Reference [34] classifies objects according to design principles, identifying three types of objects: activity-aware, policy-aware and process-aware. Objects are classified in such typologies considering awareness, representation and interaction. Awareness is a smart object's ability to understand (i.e., sense, interpret, and react to) events and human activities occurring in the physical world. Representation refers to a smart object's application and programming model. Interaction denotes the object's ability to converse with the user in terms of input, output, control, and feedback. Compared to our model, there are of course similarities with the homonym abilities named interaction and representation; in addition the awareness ability is related to our interaction and reasoning layers. Conversely, there are no abilities that match our problem solving and learning layers.

Other classifications that can be compared with our model are focused on the social interaction abilities of smart objects. Reference [39] argued that objects can be classified according to four levels of social interaction. At level 0 and 1 objects just receive or send information respectively, while level 2 can perform both tasks with a specific object and level 3 can do it with any neighbor. Atzori *et al.* [4] envision a three-step evolutionary process towards a new type of social object, from res sapiens (with limited interaction capabilities), to res agens (able to sense environment and act accordingly), to res socialis (which exhibits complex social behaviour). With reference to our model, the interaction, representation and reasoning layers are involved to manage the social relations as proposed by the authors.

Thus, we can observe that while the literature offers approaches to classify and design SPOs, some of them sharing similarities with our model, no one provides a support to design smart adaptive physical objects that are specifically focused on user support and recommendation. Most of the efforts concern the Internet of Things but they are mainly focused on the smartness of the IoT solution (often based on network services and cloud computing), not on the interactive smart support offered by the object. In this sense our model is very innovative, providing guidelines for both design and analysis - identifying a number of layers which include the features proposed by other approaches - and offering to designers methodological instruments to take decisions on the level of adaptivity, awareness and smartness of the smart object.

XII. CONCLUDING REMARKS

In the paper we introduced a conceptual model for designing smart physical context-aware, adaptive and personalized recommenders. There is in fact a growing interest for this type of smart objects and the label ''smart'' is associated to objects with very different characteristics, ranging from very simple to sophisticated ones. The model is aimed to support designers, providing multi-step guidelines for the design of an ARSPO. Each step involves a number of dimensions and choices along each dimension. In the paper we discussed the dimensions and choices, showing that each one of them corresponds to choosing different levels of sophistication and consequently different levels of smartness of the object being designed. We also discussed the relation between the level of sophistication and the complexity of the resulting objects, providing in this way a hint to find the ''best balance'' for the object being designed, given the requirements for usage, target users, problem solving goals.

The focus on physical smart objects permeated all the layers, making our approach substantially different from those focusing on smart software recommenders. The ''physicality'' of the objects, in fact, is not limited to interactional aspects (which is anyway *the* critical aspect) but influences the way an object represents and deals with itself and the world around it.

The model in the paper can be also used to analyse, classify and compare existing physical recommenders, as we exemplified in Section X. The classification is multi-dimensional and multi-faceted and thus it does not aim simply at ranking objects. Rather, it aims at providing a critical approach to evaluate pros and cons of different solutions.

The model derives from our experience in designing ARSPOs with different complexity and for different purposes. It has been distilled across time and versions of it have been progressively used in our projects, leading to the current version presented in the paper, thus providing an experimental validation of the approach.

The paper focused on the design of an ARSPO from the Computer Science point of view, without considering ''ethical'' issues'' such as privacy (considering e.g., how user's data are maintained and possibly shared), liability (concerning e.g., the recommendations provided to the user), inclusion (e.g., accessibility of the interaction modes) which will be the focus of future research.

REFERENCES

[1] A. Abbas, L. Zhang, and S. U. Khan, ''A survey on context-aware recommender systems based on computational intelligence techniques,'' *Computing*, vol. 97, no. 7, pp. 667–690, 2015.

[2] G. Adomavicius and A. Tuzhilin, ''Context-aware recommender systems,'' in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and B. P. Kantor, Eds. Boston, MA, USA: Springer-Verlag, 2011, pp. 217–253.

[3] J. Aguilar, P. Valdiviezo-Díaz, and G. Riofrio, ''A general framework for intelligent recommender systems,'' *Appl. Comput. Inf.*, vol. 13, no. 2, pp. 147–160, 2017.

[4] L. Atzori, A. Iera, and G. Morabito, ''From 'smart objects' to 'social objects': The next evolutionary step of the Internet of Things,'' *IEEE Commun. Mag.*, vol. 52, no. 1, pp. 97–105, Jan. 2014.

[5] L. Baltrunas, B. Ludwig, S. Peer, and F. Ricci, ''Context relevance assessment and exploitation in mobile recommender systems,'' *Persive Ubiquitous Comput.*, vol. 16, no. 5, pp. 507–526, 2012.

[6] P. Brusilovsky, ''Methods and techniques of adaptive hypermedia,'' *User Model. User-Adapt. Interaction*, vol. 6, nos. 2–3, pp. 87–129, Jul. 1996.

[7] P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., *The Adaptive Web, Methods and Strategies of Web Personalization* (Lecture Notes in Computer Science). Berlin, Germany: Springer-Verlag, 2007.

[8] P. Brusilovsky and E. Millán, *The Adaptive Web. Chapter User Models for Adaptive Hypermedia and Adaptive Educational Systems*. Berlin, Germany: Springer-Verlag, 2007, pp. 3–53.

[9] F. Carmagnola, ''Tag-based user modeling for social multi-device adaptive guides,'' *User Model. User-Adapt. Interaction*, vol. 18, no. 5, pp. 497–538, 2008.

[10] F. Cena *et al.*, ''Integrating heterogeneous adaptation techniques to build a flexible and usable mobile tourist guide,'' *AI Commun.*, vol. 19, no. 4, pp. 369–384, Dec. 2006.

[11] F. Cena, L. Console, A. Matassa, and I. Torre, ''Multi-dimensional intelligence in smart physical objects,'' *Inf. Syst. Frontiers*, pp. 1–22, 2017, doi: 10.1007/s1079.

[12] J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. Chi, ''Short and tweet: Experiments on recommending content from information streams,'' in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2010, pp. 1185–1194.

[13] P.-Y. P. Chi, J.-H. Chen, S.-Y. Liu, and H.-H. Chu, ''Designing smart living objects–enhancing vs. distracting traditional human–object interaction,'' in *Proc. Int. Conf. Human-Comput. Interaction*, 2007, pp. 788–797.

[14] L. Chittaro, G. Guida, C. Tasso, and E. Toppano, ''Functional and teleological knowledge in the multimodeling approach for reasoning about physical systems: A case study in diagnosis,'' *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 23, no. 6, pp. 1718–1751, Nov. 1993.

[15] P. D. Bra, G.-J. Houben, and H. Wu, ''AHAM: A dexter-based reference model for adaptive hypermedia,'' in *Proc. HYPERTEXT*, 1999, pp. 147–156.

[16] T. D. Pessemier, S. Dooms, and L. Martens, ''Comparison of group recommendation algorithms,'' *Multimedia Tools Appl.*, vol. 72, no. 3, pp. 2497–2541, 2014.

[17] T. D. Noia and V. C. Ostuni, ''Recommender systems and linked open data,'' in *Reasoning Web International Summer School*. Berlin, Germany: Springer-Verlag, Jul. 2015, pp. 88–113.

[18] L. Festinger, *A Theory of Cognitive Dissonance*, vol. 2. Palo Alto, CA, USA: Stanford Univ. Press, 1962.
[19] J. Fink and A. Kobsa, "A review and analysis of commercial user modeling servers for personalization on the world wide Web," *User Model. User-Adapt. Interaction*, vol. 10, nos. 2–3, pp. 209–249, 2000.
[20] A. P. Fiske, "The four elementary forms of sociality: Framework for a unified theory of social relations," *Psychol. Rev.*, vol. 99, no. 4, p. 689, 1992.
[21] A. Forestiero, "Multi-agent recommendation system in Internet of Things," in *Proc. 17th IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput.*, May 2017, pp. 772–775.
[22] E. van de Garde-Perik, S. Offermans, K. van Boerdonk, K.-M. Lenssen, and E. van den Hoven, "An analysis of input-output relations in interaction with smart tangible objects," *ACM Trans. Interactive Intell. Syst.*, vol. 3, no. 2, p. 9, 2013.
[23] J. Gibson, *The Theory of Affordances Perceiving, Acting, and Knowing: Toward an Ecological Psychology*. Lawrence Erlbaum Associates, 1977, pp. 67–82.
[24] N. Guarino, "Formal ontology and information systems," in *Proc. FOIS*, vol. 98. 1998, pp. 81–97.
[25] A. Guerrieri, V. Loscri, A. Rovella, and G. Fortino, Eds., *Management of Cyber Physical Objects in the Future Internet of Things: Methods, Architectures and Applications*. Cham, Switzerland: Springer, 2016.
[26] W. Hamscher, J. de Kleer, and L. Console, *Readings in Model-Based Diagnosis*. Burlington, MA, USA: Morgan Kaufmann, 1992.
[27] D. Heckmann, T. Schwartz, B. Brandherm, M. Schmitz, and A. V. Wilamowitz-Moellendorff, "Gumo—The general user model ontology," in *Proc. Int. Conf. User Model.*, 2005, pp. 428–432.
[28] M. E. P. Hernández and S. Reiff-Marganiec, "Classifying smart objects using capabilities," in *Proc. Int. Conf. Smart Comput. (SMARTCOMP)*, 2014, pp. 309–316.
[29] J. Iglesias, A. M. Bernardos, and J. R. Casar, "An evidential and context-aware recommendation strategy to enhance interactions with smart spaces," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.*, 2013, pp. 242–251.
[30] A. Jameson, "Modelling both the context and the user," *Persive Ubiquitous Comput.*, vol. 5, no. 1, pp. 29–33, Jan. 2001.
[31] S. Jang and W. Woo, "Ubi-UCAM: A unified context-aware application model," in *Proc. 4th Int. Interdisciplinary Conf. (CONTEXT)*, Stanford, CA, USA, Jun. 2003, pp. 178–189.
[32] M. Kallmann and D. Thalmann, *Modeling Objects for Interaction Tasks*. Vienna, Austria: Springer, 1999.
[33] H. R. Kim and P. K. Chan, "Learning implicit user interest hierarchy for context in personalization," in *Proc. 8th Int. Conf. Intell. Interfaces*, 2003, pp. 101–108.
[34] G. Kortuem, F. Kawsar, V. Sundramoorthy, and D. Fitton, "Smart objects as building blocks for the Internet of things," *IEEE Internet Comput.*, vol. 14, no. 1, pp. 44–51, Jan. 2010.
[35] F. Liu, C. Yu, and W. Meng, "Personalized Web search for improving retrieval effectiveness," *IEEE Trans. Knowl. Data Eng.*, vol. 16, no. 1, pp. 28–40, Jan. 2004.
[36] T. López, D. C. Ranasinghe, B. Patkai, and D. McFarlane, "Taxonomy, technology and applications of smart objects," *Inf. Syst. Frontiers*, vol. 13, no. 2, pp. 281–300, 2011.
[37] A. Matassa and R. Simeoni, "Eliciting affordances for smart objects in IoT era," in *Internet of Things User-Centric IoT*. Cham, Switzerland: Springer, 2015, pp. 77–81.
[38] G. G. Meyer, K. Främling, and J. Holmström, "Intelligent products: A survey," *Comput. Ind.*, vol. 60, no. 3, pp. 137–148, 2009.
[39] A. Moawad, V. Efthymiou, P. Caire, G. Nain, and Y. L. Traon, "Introducing conviviality as a new paradigm for interactions among it objects," in *Proc. Workshop AI Problems Approaches Intell. Environ.*, vol. 907. 2012, pp. 3–8.
[40] C. Musto, "Enhanced vector space models for content-based recommender systems," in *Proc. 4th ACM Conf. Rec. Syst.*, New York, NY, USA, 2010, pp. 361–364.
[41] J. Nielsen, *10 Usability Heuristics for User Interface Design*. Fremont, CA, USA: Nielsen Norman Group, 1995.
[42] D. A. Norman, "The psychopathology of everyday things," in *Foundations of Cognitive Psychology: Core Readings*. Cambridge, MA, USA: MIT Press, 2002, pp. 417–443.
[43] S. Parsons and M. Wooldridge, "Game theory and decision theory in multi-agent systems," *Auto. Agents Multi-Agent Syst.*, vol. 5, no. 3, pp. 243–254, 2002.
[44] S. A. Rao and P. M. Georgeff, "BDI agents: From theory to practice," in *Proc. 1st Int. Conf. Multi Agent Syst. (ICMAS)*, 1995, pp. 312–319.
[45] L. Razmerita, *Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems*, R. Sharman, R. Kishore, and R. Ramesh, Eds. Boston, MA, USA: Springer-Verlag, 2007, pp. 635–664.
[46] F. Ricci, L. Rokach, and B. Shapira, *Introduction to Recommender Systems Handbook*. New York, NY, USA: Springer-Verlag, 2011.
[47] F. Ricci, L. Rokach, B. Shapira, and B. Paul Kantor, *Recommender Systems Handbook*, 1st ed. New York, NY, USA: Springer-Verlag, 2010.
[48] S. J. Russell, P. Norvig, J. F. Canny, J. M. Malik, and A. D. Edwards, *Artificial Intelligence: A Modern Approach*. Upper Saddle River, NJ, USA: Prentice-Hall, 2003.
[49] H. R. Schmidtke, D. Hong, and W. Woo, "Reasoning about models of context. A context-oriented logical language for knowledge-based context-aware applications," *Revue d'Intell. Artif.*, vol. 22, pp. 589–608, Jun. 2008.
[50] C. Senot, D. Kostadinov, M. Bouzid, J. Picault, and A. Aghasaryan, "Evaluation of group profiling strategies," in *Proc. 22nd Int. Joint Conf. Artif. Intell.*, 2011, pp. 2728–2733.
[51] L. Tamine-Lechani, M. Boughanem, and N. Zemirli, "Inferring the user interests using the search history," in *Proc. Workshop Inf. Retrieve, Learn., Knowl. Adaptatbility (LWA)*, 2006, pp. 108–110.
[52] M. Wooldridge and N. R. Jennings, "Intelligent agents: Theory and practice," *Knowl. Eng. Rev.*, vol. 10, no. 2, pp. 115–152, 1995.

FEDERICA CENA has been an Assistant Professor with the Computer Science Department, University of Torino, since 2011. In particular, she is involved in user modeling, personalization and ubiquitous computing, and semantic Web for user modeling and adaptation. In recent years, she is mainly devoted in studying the implications of Internet of Things for user model and personalization. Her research interests include human–computer interaction and artificial intelligence. She served in the program committee for the most important conferences in her field (ACM Hypertext, ACM UMAP, and ACM IUI). She was the Program Chair at ACM UMAP 2017, the Workshop and Tutorial Chair at ACM UMAP 2016, the Publicity chair at ACM Hypertext 2014, and the Workshop and Tutorial Chair at ACM Hypertext 2014. Moreover, she organized international workshops on social adaptive semantic Web (UMAP 2010–2011) and personal informatics and quantify self (Ubicomp 2015–2017). She was an Editor of several special issues on the semantic adaptive Web for the *ACM Transactions on Intelligent Systems and Technology* journal in 2013, on personal big data for the journal *ACM Transactions on Intelligent Interactive Systems* in 2016, on gamification for the *Journal of Human-Computer Studies*, on quantified self on *Computer Journal*, and on advanced personalized mobile service on *Mobile Information System* journal in 2017.

LUCA CONSOLE received the Ph.D. degree in computer science in 1991. He is currently a Full Professor of computer science with the University of Torino, where he has been the Chair of the Department of Computer Science since 2012 and is a member of the University Boards of Directors (Senato). He has co-authored the collection of seminal works on model-based diagnosis (*Readings in Model Based Diagnoses*, with J. De Kleer and W. Hamscher) and published over 150 papers in international journals and conferences. His research interests include different areas of artificial intelligence, ranging from logical foundation of reasoning (with highly cited works on abduction and on the relations between abduction and deduction) to model-based reasoning and diagnosis (with seminal works on the foundations of abductive diagnoses and diagnosis of dynamic systems and with many application projects, especially in the automotive domain), adaptive systems, smart physical objects, and intelligent interaction. He coordinated several projects (EU projects and national/regional projects) and had research contracts from several national industries. He served in the program committee of many major conferences and has been an invited speaker in some of them, including IJCAI 1999.

ASSUNTA MATASSA received the Ph.D. degree in computer science from the University of Torino, Italy, in 2017. She was with the Intelligent User Interface Group, University of Torino. She is currently a Project Manager for IoT projects with Enel Italia. Specifically, she studied the implications of the Internet of Things in designing and developing smart objects with particular attention on objects to interact with/in urban spaces exploiting the capabilities of the human body. She published about 20 papers and organized workshops in the main international conferences on these topics. Her research interests include wearable computing, body experience, tangible interaction, urban interaction design, urban informatics, and human–computer interaction in general.

ILARIA TORRE received the Ph.D. degree. She is currently an Assistant Professor with the Department of Informatics, Bioengineering, Robotics and Systems Engineering, University of Genova, and a member of the Advisory Board of the Ph.D. School in Digital Humanities. She has been a Supervisor of several Ph.D. students and published over 70 papers in international journals and conferences. Her main research interests include intelligent user interfaces, adaptive systems, recommender systems, and ubiquitous interactive systems. Over the years, she was involved in the application of HCI and AI in various fields. She served as a Program Committee Member of a number of conferences and workshops, including ACM IUI, ACM UMAP, ACM Recommender Systems, and ACM Interact. In 2017, she has organized the Internet of Things for Active and Assisted Living Workshop at the IEEE GIobal IoT Summit 2017 and SmartLearn (Intelligent Interfaces for Smart and Ubiquitous Learning) Workshop at ACM IUI 2017. She currently leads a project on personalized privacy management in IoT social networks. She served as a Reviewer for important journals, such as *ACM Transactions on the Web*, *Pervasive Computing*, the *ACM Transactions on Interactive Intelligent Systems*.