

# A modular multi-agent framework for innovation diffusion in changing business environments: conceptualization, formalization and implementation

Simon Johanning (✉ [johanning@wifa.uni-leipzig.de](mailto:johanning@wifa.uni-leipzig.de))

IIRM, Leipzig University <https://orcid.org/0000-0002-9304-9625>

Fabian Scheller

DTU - Technical University of Denmark <https://orcid.org/0000-0002-7676-1143>

Daniel Abitz

Leipzig University <https://orcid.org/0000-0001-9603-1917>

Claudius Wehner

Leipzig University <https://orcid.org/0000-0003-3257-3279>

Thomas Bruckner

IIRM, Uni Leipzig

---

## Research Article

**Keywords:** Agent-based modeling, Agent-based product diffusion model, Innovation diffusion, Human decision-making, Socio-economic simulation, Multi-agent simulation

**Posted Date:** July 24th, 2020

**DOI:** <https://doi.org/10.21203/rs.3.rs-47351/v1>

**License:**   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

---

**Version of Record:** A version of this preprint was published at Complex Adaptive Systems Modeling on September 10th, 2020. See the published version at <https://doi.org/10.1186/s40294-020-00074-6>.

# **A modular multi-agent framework for innovation diffusion in changing business environments: conceptualization, formalization and implementation**

**Simon Johanning · Fabian Scheller ·  
Daniel Abitz · Claudius Wehner ·  
Thomas Bruckner**

the date of receipt and acceptance should be inserted later

**Abstract** Understanding how innovations are accepted in a dynamic and complex market environment is a crucial factor for competitive advantage. To understand the relevant factors for this diffusion and to predict success, empirically grounded agent-based models have become increasingly popular in recent years. Despite the popularity of these innovation diffusion models, no common framework that integrates their diversity exists. This article presents a flexible, modular and extensible common description and implementation framework that allows to depict the large variety of model components found in existing models. The framework aims to provide a theoretically grounded description and implementation framework for empirically grounded agent-based models of innovation diffusion. It identifies 30 component requirements to conceptualize an integrated formal framework description. Based on this formal description, a java-based implementation allowing for flexible configuration of existing and future models of innovation diffusion is developed. As a variable decision support tool in decision-making processes on the adoption of innovations the framework is valuable for the investigation of a range of research questions on innovation diffusion, business model evaluation and infrastructure transformation.

**Keywords** Agent-based modeling, Agent-based product diffusion model, Innovation diffusion, Human decision-making, Socio-economic simulation, Multi-agent simulation

---

Simon Johanning · Claudius Wehner · Thomas Bruckner  
Institute for Infrastructure and Resources Management (IIRM), Leipzig University, Germany  
E-mail: johanning@wifa.uni-leipzig.de

Fabian Scheller  
Energy Systems Analysis, Division of Sustainability, Technical University of Denmark (DTU)

Daniel Abitz  
Swarm Intelligence and Complex Systems, Leipzig University, Germany

## 1 Introduction

Product or service innovations are effective in creating and maintaining a competitive advantage. Thus, it is important for organizations to understand how market actors adopt innovations and engage with them, since even good innovations may fail or diffuse at a slow rate (27). For many companies, it is hard to predict how innovations will diffuse in a dynamic environment. This results in uncertainty about whether an innovation is fit to become a sustainable business model.

Their adoption by intended target groups does not only depend on the qualities of the innovation (14). Instead, it takes place within a complex social system, in which the diffusion of the respective innovations is influenced by numerous factors and mechanisms (31). Business models and innovations need to encompass the dynamics of the market setting by including the personal and mental structures of market participants, as social influences are a market-dominating factor in the innovation diffusion paradigm (18).

**Innovation diffusion (ID)** is understood as the analysis of the spread of an innovation (27). Early models on the diffusion of innovation commonly aggregated the adopters and non-adopters in singular variables (see (3) as seminal work). Over time, approaches that differentiated the actors into individual agents (so-called **agent-based models, or ABMs**) have received increasing attention. In these models, entities are modeled individually as autonomous, social, reactive and proactive agents (41). Agent actions are determined by autonomous decision-making strategies in accordance with the agents' personal objectives depending on various conditions. Phenomena in these models emerge from the behavior and interactions of the agents (18). ABMs are suited to deal with complex social and socio-ecological systems (8).

Quantitative models of innovation diffusion that account for the complexity of the modeled system are particularly suited to assist decision makers in the development of effective strategies (28). In order to model the strategies of the heterogeneous agents realistically, it is necessary to collect and analyze an extensive amount of empirical data for grounding these models (15), both theoretically and empirically (33). One promising approach for this is to employ **empirically grounded agent-based models (eABMs)** (6; 21; 19).

eABMs are gaining importance as a valuable methodology for describing ID processes (19). Since these models are primarily applied to reflect real market issues (33), papers with real-world case studies to support decision makers are becoming more popular (34). This is also due to their capability of explaining and reproducing observed complex non-linear diffusion patterns through relatively simple local micro-level interactions (18). Case-based applications *“have an empirical space-time circumscribed target domain.”* (5). They are usually built *“to provide forecasts, decision support, and policy analysis [...]”* (19), showing that the application domain of eABMs for ID is very versatile. Similarly, existing eABMs differ extensively in their design and grounding and therefore also in their predictions and conclusions (34; 44).

Despite the individual differences among models, many eABMs share a lot of commonalities. However, as (4) stated, in most cases models are developed primarily without regarding existing approaches and shared common structures, resulting in a lack of "[...] a clear foundation of agreed-upon approaches and libraries that offer a baseline for problem solutions that characterize other modeling fields.". The selection of modeling aspects is often ad hoc (12) without any defensible rationale (34). In the current culture, papers furthermore commonly fail to document critical methodological details (34). In line with this, the problem this article addresses is:

*Problem statement:*

Assessing the diffusion of new products is a business-critical activity. Due to the lack of a common foundation of eABM on ID, model construction is often adhoc, as hardly any generic framework for ID assessment exists.

A generic framework for ID assessment that integrates and extends modeling approaches of eABMs permits modelers to compare and evaluate specific model mechanisms with little effort (29). This allows the evaluation of modeling decisions and thus the identification of the model most appropriate for the subject at hand. Individual, context-specific models of concrete complex phenomena require extensive diligent data collection (empirical grounding), as noted above. The foundational structure of the models (i.e. the components, relationships and model dynamics), on the other hand, needs to be grounded on theory or existing models (theoretical grounding). The latter is particularly important for systems that aim at supporting various models and modeling contexts, most of all those aspiring to be generic frameworks.

While empirical grounding is specific to the modeled context, theoretical grounding is a more overarching requirement that can be addressed by basing the model context on existing, proven models. This property can be well-encompassed by an appropriate framework, resulting in the following requirements:

*High-level requirements:*

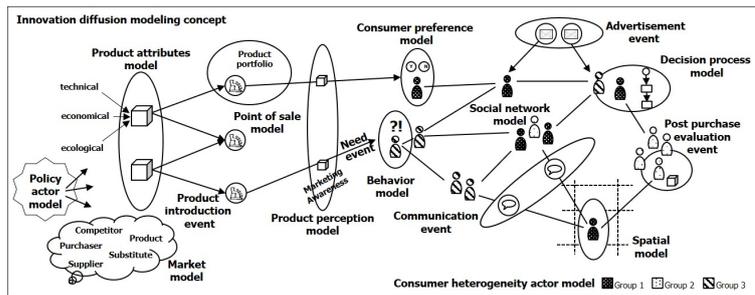
- R1 The framework must allow to depict the large variety of model components found in existing models.
- R2 The framework must be flexible and modular to address the diversity of innovation diffusion models.

It should be noted that the framework needs to furthermore be extensible to account for future ID models. However, through the flexibility and modularity, this requirement is given implicitly through R2. Based on the (high-level) requirements R1 and R2, a solution addressing the problem stated above is characterized as follows:

*Solution objective:*

A solution needs to provide a flexible, modular and extensible common description and implementation framework that allows to depict the large variety of model components found in existing models in order to provide a theoretically grounded description and implementation framework for empirically grounded agent-based models of innovation diffusion.

Achieving this objective yields an integrated model framework suited to incorporate decisive components of proven models, as well as providing a formal model and a software implementation for existing and novel applications. As part of a research endeavor on infrastructure modeling, the framework is coined *Integrated Resource Planning and Interaction* (or short: IRPact). IRPact is based on a conceptual overview of various model components and dynamics of (29). A summary of major aspects is depicted in Figure 1. From a scientific point of view, the presented eABM framework aims to analyze the ID process interrelations between customer behavior and product attributes. The framework is intended to both give researchers orientation on model development and to reduce construction and conceptualization time by providing a common framework. It furthermore strives to increase model and result robustness by allowing model component and parameterization evaluation. In practice, the framework can help decision makers to evaluate business model innovations in a fast-changing environment and thus develop a sustainable business strategy.



**Figure 1** Illustrative representation of identified agent and product entities, model dynamics and simulation foundations (based on the results of (29))

This article is organized along the process of the research it describes: Section 2 reviews and synthesizes existing components of eABMs for ID analysis and derives the conceptual requirements of the framework. Subsequently, the mathematical eABM framework is outlined in Section 3. The model description serves to describe the design of a versatile eABM framework to evaluate ID processes under different system conditions. Section 4 elaborates on the software implementation of the framework serving to facilitate the flexible configuration of the developed eABM. In Section 5 the developed model and the development process are discussed and further work is addressed.

## 2 Component analysis

In order to theoretically ground the framework and address requirement **R1**, this section identifies **component requirements (CRs)** it needs to fulfill. It builds upon a systematic review of eABMs as presented in (29).

The identified concepts and inferred modeling requirements are grouped by components representing **model foundations**, **products**, **mental structures**, and **agents**. An illustrative high-level illustration of the interplay of the identified and integrated model components is given in Figure 1. A summary of each of the requirements is given in Table 1.

Table 1: Summary of the component requirements for the multi-agent framework.

Nr.	Component	Design requirement
CR 1	Temporal model	Specification of temporal model context
CR 2	Process model	Management of stages and dynamics of the simulation
CR 3	Spatial model	Appropriate level of spatial representation
CR 4	Network model	Possibility to depict social network as graph
CR 5	Network model	Configuration of different network topologies
CR 6	Network model	Flexibility for numerous forms of social ties
CR 7	Product entity	Product innovation with variety of product qualities
CR 8	Product entity	Manipulation of the product innovations in the simulation
CR 9	Perception	Duality of true qualities and subjective perceptions
CR 10	Perception	Dynamic socio-economic product perceptions
CR 11	Perception	Possibility for observing adopted products
CR 12	Perception	Consideration of consumer unawareness of products
CR 13	Preferences	Agent-specific preferences by environmental and social values
CR 14	Preferences	Linkage of product qualities and preferences for evaluation
CR 15	Decisions	Integration of various dynamics and attributes
CR 16	Decisions	Highly flexible and heterogeneous decision processes
CR 17	Decisions	Initiation of the decision process through a need event
CR 18	Communications	Communication component to address social perception
CR 19	Communications	Adaptable and integrated communication channels
CR 20	Communications	Communication with influence on behavior or belief
CR 21	Communications	Heterogeneous and spatial communication schemes
CR 22	Advertisements	Provision of mass media and advertisement capability
CR 23	Advertisements	Perception or preference change of agents
CR 24	Consumer agent	Consumer agent with individual attributes
CR 25	Consumer agent	Intra-homogenous and inter-heterogeneous agent groups
CR 26	Company agent	Company agent with product-relevant consumer influence
CR 27	Company agent	Proactive product management and marketing
CR 28	Sales agent	Sales agents with product portfolio
CR 29	Sales agent	Proactive behavior and purchase processes in spatial context
CR 30	Policy agent	Policy agent with with re- or proactive policy management

### 2.1 Model fundamentals

ABMs are grounded on several modeling strategies related to (physical) fundamentals, in particular time, space and processes. These strategies are important for the implementation as well as for processing logic. In particular, to depict temporal dynamics, the temporal model context needs to be an integral part of ID models (CR 1). While nearly all models employ discrete timing schemes (time passes in some temporal unit, and all or some agents act in each step), (36; 26) implement continuous time scales. Yet, **temporal models** are barely ever discussed explicitly (with the notable exception of (18)). The fo-

cus of many models is on process modeling, while temporal progress is only implicitly modeled.

Thus, many temporal aspects of these models are situated within the **process model** and the temporal model is described slenderly (1; 2; 17). The process model according to (36) governs the stages of the innovation decision process (with at least the stages of awareness, trial and adoption) of (27), and specifies the execution of processes (for discrete temporal models) or actions between events (for continuous temporal models). In the framework it thus needs to manage adoption stages and simulation dynamics (CR 2).

In addition to temporality, spatiality is an important characteristic in a range of different eABMs (24; 35; 37; 32). It allows to explicitly integrate the geographic location of entities and needs to address how geometry and the relative positioning of entities is operationalized. Due to the different requirements, an appropriate level of spatial representation needs to be reflected (CR 3).

Providing the infrastructure for communication and influence dynamics and utilizing the spatial characteristics, the **social network** can be seen as an interconnecting system (10; 11; 22). The social network describes the connection between agents and is a crucial aspect of several ID models. Frequently, the network is formalized as a directed or undirected graph with model entities for nodes and communication channels for edges (CR 4). The network structure can be given explicitly such as scale-free (10), purely local (22) or small-world (35) network topology. Thus, different network topologies need to be possible (CR 5). A social network can stand for a number of social ties, sometimes even several ones within one model as described in (20), where friendship and trading relationships are incorporated. It thus needs to provide flexibility for numerous forms of ties (CR 6).

## 2.2 Product modeling

Obviously for ID models, the **product entities** analyzed are of fundamental importance. Due to the heterogeneity of the investigated models, technologies show a large variety of attributes ranging from technical parameters (42; 32) through ecological characteristics (39; 23) to cost-related parameters (13; 23). The framework must allow to for depicting this variety (CR 7). While in most models products are static entities within the time frame of adoption, some models require certain dynamics of the modeled products. In this context, (31) incorporates products that enter the simulation at a later time (market introduction) or become unavailable for adoption before the end of the simulation (product discontinuation), requiring a dynamic product set (CR 8).

## 2.3 Mental structures

To enable the use of models sensitive to cognitive modeling, it is important to depict not only the true qualities of innovations but also the **perception**

of qualities. Perceptual aspects of products identified in the literature are ecological aspects (13), social perception (36) or substitute availability (39), which intends to model acquiring knowledge about qualities and existence of products. This requires a duality of product's true qualities and agents subjective perceptions (CR 9). To address cognitive modeling, eABMs need to address incomplete information, cognitive distortions and (subjective) perception. Thereby perceptions are not always static and can change over time (CR 10). Additionally, an interaction between perceptions and the true nature of products can also be seen by (18), where the true attributes of the products can only be assessed with a post-purchase evaluation of the adopted products (CR 11).

Another aspect of incomplete information is knowledge of the existence (awareness) of products (CR 12) since individual decision-making is based on the on the basic awareness of the technology (27).

A further source of heterogeneity in eABMs is seen in preferences. Analyzed models express **preferences** as ecological aspects of products, such as environmental concerns (23) or certain aspects of social perception (36; 35), requiring differentiated agent-specific value-specific preferences (CR 13). Models further incorporate attitudes relating to preferences in order to be used by decision processes (42), e.g. for weighing product aspects (13). Thus, preferences need to link to product qualities in order to relate values to the evaluation of products (CR 14).

Being a source of consumer heterogeneity and an essential component of ID, **decision making** is arguably one of the most important aspects of ID modeling. It can be understood as the cognitive processes employed for deciding on a product. Decision making is influenced by a variety of various aspects, e.g. mass media and advertisement, communication, product attributes and decision processes (30). At the same, social influence is also an important aspect in decision making (16), which might also be coupled with spatial factors and serve as another source of heterogeneity (31). Decision processes thus need to integrate a range of dynamics and attributes (CR 15). Modeling the decision process is grounded in various decision theories or in utilitarian approaches (38; 23; 36; 30; 1; 24). Other approaches include game theory (42) or artificial neural networks accounting for emotions in information processing (40). Thus, an innovation diffusion eABM framework must allow for a large variety and flexibility of decision processes (CR 16). Additionally, a decision to adopt is always motivated by the non-fulfillment of some need of the agents to adopt. The initiation impulse and the corresponding intensity of the need requirement therefore also needs special attention as shown by (36) (CR 17). Thereby, different decision theories are conceivable (23; 36; 1; 24).

Moreover, word of mouth is an important phenomenon featured in ID eABMs. Due to the multitude of possible communication contents, numerous model aspects fall into this category. Arguably, the most important function of communication is to foster social perception (36; 9; 13), which needs to be covered by the framework (CR 18). One set of channels mentioned in literature is connected to advertisement and mass media (38; 36; 40). Others are

communication channels between consumer agents, taking into account the social network (36; 24; 40). The model described in (30) uses a more complex channel, which can be modeled by mass media and advertisement, communication, product attributes and the decision process. In order to capture this variety and complexity, communication channels need to be adaptable and be able to integrate this variety (CR 19). For models exemplifying behavior and belief change (40), the framework needs to allow to have behavior and beliefs be mediated through communication within a social network (CR 20). Further model aspects touching upon communication are learning, social norms, mental representation and personal attributes such as social influence and the number of peers to communicate (40). Communication behavior further shows heterogeneity by social and spatial factors (31). Due to the plurality of communication forms and contents, **communication modeling** thus needs to allow for heterogeneous and spatially-sensitive communication processes (CR 21).

In addition to communication between consumers, (38) imply a corporate entity using mass media. The publications of (36), (40) and (7) make use of channels related to advertisement and mass media, while (30) model the interplay of mass media and aspects of product attributes and decision processes. The framework thus needs to allow for mass media and advertisement (CR 22). More abstractly, an ID model needs to address information and advertisement modeling, particularly how a range of information and advertisement processes take place. This includes changing the perception or preference of an agent, taking into account the nature of a product for information and some desired value for advertisement which is proactively sought instead of pushed on agents (CR 23).

## 2.4 Agent modeling

Innovation diffusion is primarily concerned with adoption decisions of agents or stakeholders, making **consumer agents** (e.g. individuals or households) the center of these models. They are characterized by a large range of personal, economic and social attributes. Examples for this are subjective norms (16) or characteristics within the socio-economic coordinate system (40), such as age, salary, years of ownership, average behavior (13), or innovativeness (30), necessitating individual attributes for these agents (CR 24). Consumer heterogeneity has been explicitly mentioned by various models based on heterogeneous social structure (32; 23; 13; 22; 35). This heterogeneity is not only seen in individual aspects, but also those characterizing whole groups of agents and their relationship with other groups. Thereby, the grouping is often done on a socio-economic or psycho-sociological basis (such as lifestyle groups, cognitive involvement in decision processes (31)), requiring groups of similar agents (intra-homogeneity) that are distinct from other groups (inter-heterogeneity), resulting in component requirement CR 25.

In addition to consumer agents, **company agents** have been modeled in different eABMs, be it indirectly as in (1) and (42), or directly as brand agents (30). In this role, corporation entities interact with the consumer agents to exert product-relevant consumer influence (CR 26) and prompt a decision process. Furthermore, active product management such as strategic alignment (1), suppliers influence (43; 20) and competitors influence was modeled and should be possible to implement (CR 27). In (30), corporate entities are modeled directly (and endogenously) as brand agents which as sales agents also can handle different product portfolios of different companies (CR 28). This is even more the case for models in which the point-of-sale (POS) is explicitly depicted as physically distinct **sales agent** with a range of products, product availabilities and prices, and a purchase process as implemented by (36). Similar to other agents, and as outlined in (30; 1; 43; 20) for company or brand agent a sales agent should furthermore exhibit proactive behavior, resulting in the need for proactive behavior and purchase processes that optionally take place in a spatial context (CR 29).

Additionally, a **policy agent** is used both endogenously and exogenously in existing eABMs, e.g. in (42) and (2). While in the first case rules and regulations are set by authorities, in the second case administration tasks are aggregated by a policy agent. In order to depict this, a suitable framework should allow for the pro- or reactive management of existing regularities and policies (CR 30).

### 3 Framework modeling

This section presents the modeling approach and the formal definition of model components based on the component requirements described in Section 2, as summarized in Table 1. Starting from the **fundamental structures** that other model components are embedded in (**time** and **space**, the **executed processes** and the structure of the **social network**), the fundamental object of the model, the **product**, is defined. In order to introduce all crucial components agents depend on, **mental structures** (**perceptions**, **preferences**, **decision processes** and **communication**) are introduced before a comprehensive overview of the different **agents** (**consumers** and their **groups**, **company agents** and **points-of-sales**, as well as the **policy agent**) is given.

#### Modeling approach

As mentioned in the **introduction**, the modeled framework needs to be flexible and modular (R2) in order to be extendable. This was addressed through a modeling process following a two-tiered abstraction approach. Possible entities or mechanics are integrated into sets and entities are defined by tuples of members of these sets. These sets are often called a scheme, of which a concrete specification appears as an element in the tuple defining a component.

Equivalently, for similar entities with common characteristics, a structure understood as the group is used. Entities associated with them (as derived instances) are formalized through an association function.

The requirement for allowing a large variety of model components (R1) has been detailed into 30 component requirements that guided the modeling process. Before the components are detailed below, a short summary of the model is given and visualized in Figure 2.

In the modeled processes, **consumer agents** belonging to **consumer agent groups** adopt **products** through **decision making processes** based on consumer **perceptions** and **preferences** that can change over **time**. Perceptions of products consumers are **aware** of are shaped through **communication** with other consumer agents in their **social network** and **advertisement** from **corporate agents** that **manage** these products. Consumer agents are furthermore exposed to products of various **product groups** through **point of sales** that can be embedded in the **spatial** model. A **policy agent** can further restrict the product landscape.

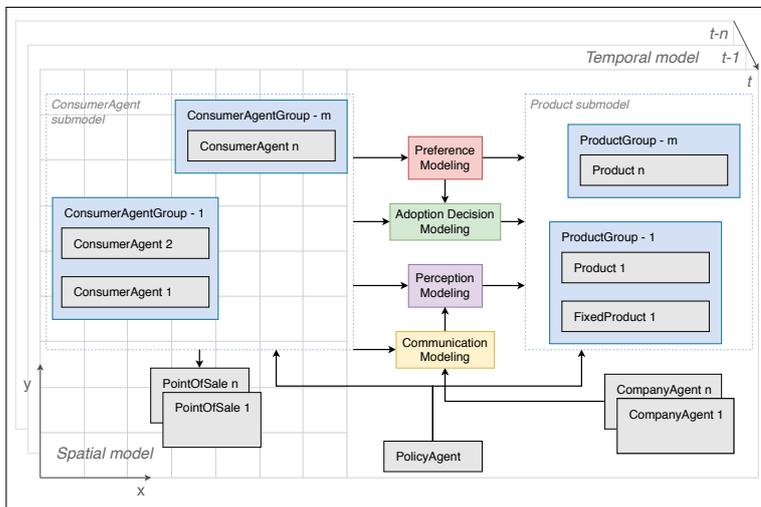


Figure 2 Overview of IRPact framework components.

### 3.1 Model fundamentals

#### 3.1.1 Temporal modeling

In addition to the specification of the temporal model context (CR 1), a more general framework needs to enable the implementation of different timing schemes. The temporal model further serves as a frame of reference.

Different time models allow the system to follow different timing regimes. This comprises the discrete and continuous time model. In general, discrete time models operate in steps that advance the time of the simulation step by step (usually  $T \subset \mathbb{N}_0$ , with  $T$  as the set of time steps in the simulation). In this, only model dynamics corresponding to these time points are valid. For continuous times regimes all time points prior to the simulation end time are valid and the mechanism governing simulation time is more event-driven. This is described by  $T \subset (t \in \mathbb{R}_+ | t \leq t_{end})$ , with  $t_{end} > 0$  the end time of the time horizon of interest.

While the timing of the continuous model is marked entirely by events, through their relation to the discrete time, events are used in the discrete setting as well. Events bind dynamic model semantics to the temporal evaluation of their desired effect. This touches on a number of processes, such as the **market introduction or discontinuation of a product**, **communication events**, the rise of **consumer needs** and **post-purchase evaluations**. These are detailed below in the respective subsections.

### 3.1.2 Process modeling

In order to manage the stages and dynamics of the adoption process (CR 2), IRPact uses a process model. It allows for various process models, but is primarily intended with adoption stage processes in mind. In a stereotypical implementation it consists of a knowledge, persuasion, decision, implementation and confirmation step, but is open to different process models as well. For the continuous case it works with events corresponding to the phases and specifies actions with regards to product readoption.

The process model specifies how agents are processed, and is based on an event scheduler, especially in the case of continuous temporal models.

### 3.1.3 Spatial modeling

The **analysis of existing models** showed that the spatial aspects of the simulation should be captured through a **spatial model** with an appropriate level of spatial representation (CR 3).

Whereas (spatial) agents are positioned through their coordinates *loc*, their positioning towards the geometry of other spatial model aspects such as the model border is governed by the spatial border map  $\mathfrak{s}bm : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{B}$ . This map describes whether a (two-dimensional) coordinate *loc* falls within the borders of the model.

Entities within the simulation are further associated with a spatial distribution for the geographical initialization of the model. These entities are thus assigned a position in the spatial model  $\mathfrak{S}$  according to their configured spatial distribution (see the respective agents for more detail).

### 3.1.4 Social network modeling

As mentioned in section 2, communication needs to respect the social network of agents. The formulation of the social network as a graph addresses both component requirement CR 4 (possibility to depict social network as graph) with different network topologies (CR 5) and gives the social network large generality. The social network is modeled as a dynamic directed weighted (multi-) graph with agents as nodes and connections (ties) between agents as edges. Central to the social network is the structure of communication channels between consumer agents through different media (such as communication flow, information flow, friend-of-relations etc.).

The social network is modeled as a 4-tuple  $SN = (G, \mathfrak{w}, \text{ews}, \text{tms})$ . At the core of the social network is the social graph  $G = (AN, E)$ , describing the relationships between agents as nodes  $r, o \in AN$  through directed edges  $e = (r, o, m) \in E$  (from node  $r$  to node  $o$ ), where each edge is associated with a medium  $m \in M$ . A node  $r$  is associated with a consumer agent  $c$  through the consumer agent social graph association mapping  $r = \text{csgam}(c)$ .

The edges are further associated with different interactions through the medium<sup>1</sup>, as specified in other model components representing different qualities of information flow. Similarly, the strength of the ties is formalized through the edge weight function  $\mathfrak{w} : E \times T \rightarrow \mathbb{R}_{\geq 0}$ . This allows for rich interpretation and enables dynamic behaviour and heterogeneous edge weights. The multitude of media and their dynamic nature provides a wide variety of agent connections, addressing requirement CR 6.

In order to describe the dynamics of the social network, social networks furthermore feature edge-weight schemes  $\text{ews} \in EWS$  that describe the temporal change of the weights  $\mathfrak{w}$ , as well as topology manipulation schemes  $\text{tms} \in TMS$ . These schemes describe how the connections of the social network (i.e. the edges between nodes  $r, o \in AN$ ) change through model dynamics by establishing new links or shedding existing ones.

## 3.2 Products

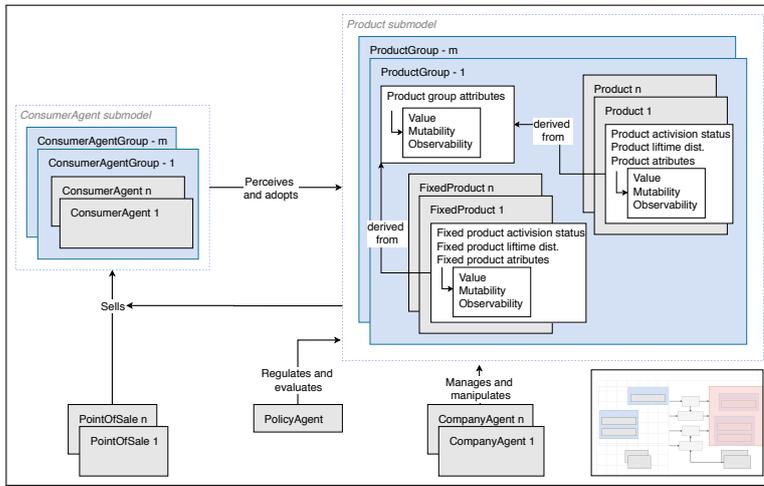
In order to allow for rich variety of product qualities (CR 7), products follow a modeling approach on two abstraction layers: product groups as (abstract) blue-prints of similar products and concrete product instances belonging to different product groups. Products contain a set of **product attributes**, describing various qualities of products and are organized in **ProductGroups** through the Product Group Association Map:  $\text{pgam} : P \rightarrow PG$ , associating products  $p \in P$  with their product groups  $pg \in PG$ . Products can be parameterized either through stochastic product initialization (meaning that product attributes are assigned using a probability distribution) or can be configured as **fixed products** with set values.

A product  $p \in P$  is defined as a 3-tuple

<sup>1</sup> Such as information exchange, value exchange, trust, marketing information etc.

$$p_t = (PA_{p,t}, pas_{p,t}, \mathcal{PLD}_p), \text{ with}$$

- $PA_{p,t}$  Set of product attributes at time  $t$  with entries  $pa_t \in PA_{p,t}$ .
- $pas_{p,t}$  Product activation status  $pas_{p,t} = \mathbf{apm}(p, t)$  (as defined through the adopted product map  $\mathbf{apm}$ , indicating whether the product is already introduced and not yet discontinued at a given time  $t$ ).
- $\mathcal{PLD}_p$  Product lifetime distribution, specifying how long a product can be used upon adoption before it is removed from the adopted products  $ap_{c,t}$  of consumer agents  $c \in C$ .



**Figure 3** Product related entities of IRPact modeling components.

### 3.2.1 Product attributes

Product attributes  $pa_t \in PA_t$  describe the qualities of products quantitatively. For this, the product attribute value  $pav_{pa,t}$  holds numerical values on a number of quality dimensions (one for each product attribute). The product mutability  $pam_{pa}$  indicates whether  $pav$  is allowed to change over time and the observability  $pao_{pa}$  describes to what extent its true quality can be assessed by an agent.

A product attribute  $pa_t$  at time  $t$  can thus be described as a 3-tuple

$$pa_t = (pav_{pa,t}, pam_{pa}, pao_{pa}) \in PA_t, \text{ with}$$

$pav_{pa,t}$	Product attribute values $pav_{pa,t} = \mathbf{pavm}(pa, t)$ described through the product attribute value map $\mathbf{pavm} : PA \times T \rightarrow \mathbb{R}_{\geq 0}$ at simulation time $t$ .
$pam_{pa}$	Mutability defined by the product attribute mutability map $pam_{pa} = \mathbf{pamm}(pa) : PA \rightarrow \mathbb{B}$ .
$pao_{pa}$	<b>Observability</b> $pao_{pa} = \mathbf{paom}(pa)$ is given by the product attribute observability map $\mathbf{paom} : PA \rightarrow [0, 1]$ .

### 3.2.2 Product groups

As an abstraction of similar products with individual differences, product groups serve to balance homogeneity and heterogeneity through the use of probability distributions for the description of their attributes:

A product group  $pg \in PG$  is formalized as an 8-tuple<sup>2</sup>

$pg = (PGA_{pg}, PPG_{pg}, EPG_{pg}, FP_{pg}, pgn_{pg}, SP_{pg}, odp, \mathcal{P}\mathcal{L}\mathcal{D}_{pg})$ , with

$PGA_{pg}$	Set of product group attributes associated with $pg$ .
$PPG_{pg}$	Set of prerequisite product groups for the product group.
$EPG_{pg}$	Excluding product groups.
$FP_{pg}$	Set of <b>fixed products</b> of product group $pg$ .
$pgn_{pg}$	Set of needs derived products fulfill.
$SP_{pg}$	Standard product for the product group $pg$ .
$odp$	overwrite decision process $odp = \mathbf{odpm}(pg)$ .
$\mathcal{P}\mathcal{L}\mathcal{D}_{pg}$	Product lifetime distribution (default) used for products derived from this product group.

### 3.2.3 Product group attributes

Product group attributes formalize the quality of product groups. Instead of assigning a scalar value on the quality dimension of product attribute values, product groups describe these through probability distributions. Observability and mutability, however, are constant over all ProductAttributes derived from this ProductGroupAttribute.

Analogous to product attributes, a product group attribute is formalized as a 3-tuple

$pga = (\mathcal{P}\mathcal{G}\mathcal{A}\mathcal{V}\mathcal{D}_{pga}, pgam_{pga}, pgaopga)$ , with

$\mathcal{P}\mathcal{G}\mathcal{A}\mathcal{V}\mathcal{D}_{pga}$	Distribution the product attribute values are drawn from via realizations of the random variable $X_{PGAVD_{pga}}$ .
$pgam_{pga}$	Product attribute mutability $pgam_{pga} \in \mathbb{B}$ .
$pgaopga$	Attribute observability $pgaopga \in [0, 1]$ of derived product attributes.

<sup>2</sup> For the sake of brevity and legibility, not all aspects of product groups are explained here. For a full model description see the model description on [CoMSES Computational Model Library](#) and [github](#).

The mapping between the product group attributes and the attributes belonging to them is done through the product group attribute product attribute mapping  $\mathbf{pgapam} : PA_t \rightarrow PGA$ . With the exception of **fixed products**, the respective initial value of the product attribute  $pav_{pa,0}$  is determined through  $X_{PGA VD_{pga}}$  with  $pga = \mathbf{pgapam}(pa)$ .

### 3.2.4 Fixed products

In contrast to assigning attribute values through a stochastic process, fixed products  $fp \in FP = \bigcup_{t \in T} FP_t$  exhibit predetermined initial values  $pav_{pa,0}$  for the respective  $PA_{fp,0}$ . Fixed products are generally used to parameterize scripted events, that is the **market introduction or discontinuation** of a product explained below, allowing the manipulation of the simulation product set, addressing CR 8.

The initial relation of consumer agents and existing fixed products is described through the respective fixed product awareness distributions  $\mathcal{FPA}_{cag}^{fp}$  (specifying how **awareness** about  $fp$  is initially distributed in  $cag$ ) and the initial fixed product adoption distributions  $\mathcal{IAPA}_{cag}^{fp}$  (describing how the adopters of  $fp$  at the beginning of the simulation are distributed), as seen **below**.

Fixed products  $fp \in FP$  are likewise formalized as a 3-tuple, only they bypass the stochastic generation step:

$$fp_t = (PA_{fp,t}, pas_{fp,t}, \mathcal{PLD}_{fp}), \text{ with}$$

$PA_{fp,t}$	Set of fixed product attributes at time $t$ with entries $pa_t \in PA_{fp,t}$ .
$pas_{fp,t}$	Product activation status $pas_{fp,t} = \mathbf{apm}(fp, t)$ in analogy to other products.
$\mathcal{PLD}_{fp}$	Product lifetime distribution that can differ from other products in the product group, but usually corresponds to those used for stochastically initiated products.

## 3.3 Mental structures

### 3.3.1 Perception modeling

The primary purpose of the perception sub-model is to enable the duality of true qualities and subjective perceptions (CR 9). While the true qualities of products are already modeled through **product attributes**, the framework requires individual perceptions of these qualities. These further need to be allowed to change over time to allow for CR 10. The **observation of adopted products** (CR 11) and **(un-)awareness** about products (CR 12) is described below.

Perception is modeled in a two-tiered fashion in the framework: perceptions of the values of a product attribute  $pa$  for consumer agent  $c$  at time  $t$  are described through the perceived product attribute value map  $\mathbf{ppavm} : C \times PA \times T \rightarrow \mathbb{R}_{\geq 0}$ .<sup>3</sup> Their temporal dynamics abstract some of the technicalities of the  $\mathbf{ppavm}$  away through perception schemes  $ps \in PS$  describing the initial status of the  $\mathbf{ppavm}$ , e.g. as the perceived product group attribute value distributions  $\mathcal{P}\mathcal{P}\mathcal{A}\mathcal{V}\mathcal{D}_{cag}$  associated with consumer agent group  $cag$ <sup>4</sup>, and the temporal dynamics (i.e.  $(\mathbf{ppavm}(c, pa, t) | \mathbf{ppavm}(c, pa, \hat{t}), \hat{t} < t)$ ).

Perception schemes thus describe how perception-forming events are integrated over time, and a single value  $\mathbf{ppavm}(c, pa, t)$  is calculated at a given time  $t$ .

The association of the product group attributes and the respective perception schemes is formalized through the product perception scheme mapping  $\mathbf{ppsm}_{cag} : PGA \rightarrow PS$ .

### Product observation

The observation of adopted products (CR 11) is realized through perception opportunities after adoption of the respective product (so-called post-purchase evaluation). Post-purchase evaluations add a product perception to the consumer's product attribute perception with a weight based on the **observability** of the product and the true product attribute value at observation time  $\hat{t}$ :

$$\mathbf{ppavm}(c, pa, t) | (\mathbf{ppavm}(c, pa, \hat{t}), pa o_{pa}, pav_{pa, \hat{t}})$$

To what degree this changes the perception of the consumer agent depends on the respective perception scheme  $ps$  used.

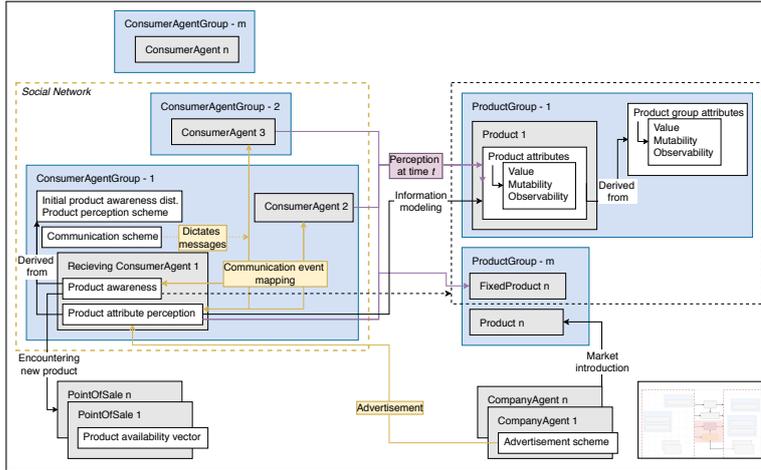
### Product awareness

While product attribute perception captures the imperfect information consumers have of the true value of a product attribute, product awareness describes whether a consumer is aware of a product (if a product is on the market), addressing component requirement CR 12. Knowledge of a product's existence is modeled through the product awareness map  $\mathbf{pavm} : C \times P \times T \rightarrow \mathbb{B}$ , describing whether  $c$  is aware of  $p$  at time  $t$  ( $\mathbf{pavm}(c, p, t) = \text{true}$ ). It is initially parameterized through the product awareness distributions  $\mathcal{P}\mathcal{A}\mathcal{W}\mathcal{D}_{cag}^{pg}$ , and is specific for each product and consumer agent group. Its value is *true* after product encounter, e.g. through **communication about the product** by another consumer agent, company originated consumer agent messages or by encountering a product in a **POS**.

An overview of the described perception modeling is given, inter alia, in Figure 4.

<sup>3</sup> Which is a partial function, since it is not defined for values of products a consumer is not **aware** of (that is  $\mathbf{ppavm}(c, pa, t) = \text{undef} \Leftrightarrow \mathbf{pavm}(c, p, t) = \text{false}, p : pa \in PA_p$ ).

<sup>4</sup> As  $\mathcal{P}\mathcal{P}\mathcal{A}\mathcal{V}\mathcal{D}_{cag}$  depends on the perception scheme associated with  $cag$ , it only indirectly defines consumer agent groups and is not part of their constituting tuple.



**Figure 4** Perception, communication and information modelling with the relevant components of the IRPact framework.

### 3.3.2 Preference modeling

Preference modeling addresses agent-specific preferences based on environmental and social values (CR 13) and needs to allow to link product qualities and preferences for product evaluation (CR 14). Moral or ethical dimensions of consumer agents are modeled through values  $v \in V$  and the importance (strength) agents assign to them, as a numerical value. Preferences are formalized through the preference map  $\mathbf{pm}_c : V \times T \rightarrow \mathbb{R}_{\geq 0}$  assigning a numerical value to the strength of value  $v \in V$  for consumer  $c$  at simulation time  $t$ . The relation between preferences and product attributes is modeled as a weighted map, the (temporally static<sup>5</sup>) *product attribute value preference mapping*  $\mathbf{pavpm} : PA \times V \rightarrow \mathbb{R}_{\geq 0}$ , to be used in product evaluation in the purchase decision process.

### 3.3.3 Adoption decision modeling

At the core, simulating product adoption is about modeling the decisions of consumer agents between suitable products. The decision processes involved in this should allow to connect advertisement, communication, product attributes and decision processes (CR 15). They should further be highly flexible (CR 16), and allow for taking into account social and spatial factors.

In IRPact, this is formalized through an abstract decision process  $d$  associated with a consumers' **agent group**, representing the cognitive processes they employ for deciding between suitable products. From the perspective of the

<sup>5</sup> with the potential exception of the market introduction of new products, where a new mapping is added.

model, *needs* the products' *ProductGroup* fulfills are satisfied with the adoption of the respective product and decision processes are initiated through a **need event** (CR 17).

### Decision process modeling

Formally, decision processes are specified by the product adoption decision map  $\mathbf{padm} : C \times \mathcal{P}(P) \times T \times D \rightarrow P$ ,  $\mathbf{padm}(c, P_{pot}, t, d) = p$ , describing the taken product adoption for product  $p$  of agent  $c$  with the eligible potential products  $P_{pot}$  at time  $t$  for decision process  $d$ . For decision modeling where the state of agents in the social network of an agent or other system aspects are used, this is extended to the extended product adoption decision map:  $\mathbf{epadm} : \mathfrak{M} \times T \rightarrow P$ , with  $\mathfrak{M}$  representing the model as a tuple of all its components.

The decision process is triggered when a **need event** is processed and is thus governed by the **process model**, often depending on the internal state of the agents and certain environmental parameters. Decision processes can be executed at a time depending on the modeled system dynamics, governed by the process model as well.

When a product  $p$  is adopted (i.e.  $\mathbf{padm}(c, P_{pot}, t, d) = p$ ), it is added to the set of adopted products  $ap_{c,t} = \mathbf{pam}(c, t)$  of agent  $c$  for the lifetime of the product adoption, as drawn from the respective distribution  $\mathcal{PLD}_{pga}$  at adoption time. This is formalized through the partial product adoption lifetime map  $\mathbf{palm} : C \times T \times P \rightarrow \mathbb{B}$ , describing whether an adopted product is operational (i.e. in a state to satisfy the associated need) at a given time, and is related to  $\mathbf{pam}(c, t)$  by the following:

$$\mathbf{palm}(c, t, p) = \begin{cases} 1 & p \in \mathbf{pam}(c, t) \wedge t \in [\hat{t}, \hat{t} + x_{p,c,\hat{t}}^{pl}] \\ 0 & \text{else} \end{cases}$$

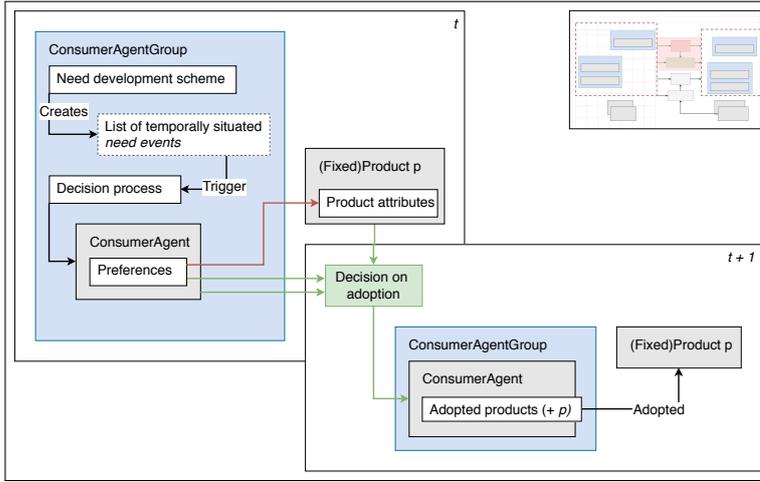
with  $\hat{t}$  being the time the product is adopted<sup>6</sup>, i.e.  $\hat{t} = \arg \min_t p \in \mathbf{pam}(c, t)$ , and  $x_{p,c,\hat{t}}^{pl}$  being the realization of the random variable  $X_{pl}^{c,t}$  corresponding to the product lifetime distribution  $\mathcal{PLD}_p$  for product  $p$  adopted by consumer  $c$  at time  $\hat{t}$ .

The entire process of adaption is showcased in Figure 5.

### Needs modeling

Needs  $n \in N$  derive their semantics from products satisfying them, decision processes, and the process describing how they arise. The latter is called a *need development scheme*  $NDS_{cag}$ , associated with *ConsumerAgentGroup*  $cag$ ,

<sup>6</sup> To not make it overly complicated, although technically wrong, the authors chose to express it like this, although naturally the same product could be adopted several times after it expired; This is respected in the implementation, but omitted here for clarity.



**Figure 5** Preference and adoption decision modeling process of a single consumer agent.

which is invoked by the process model to create an ordered list of needs, operationalized as need events. These are temporally situated events that trigger decision processes, which are governed through the process model.

Need development schemes differ greatly by their dynamics, and are specified by what happens with products if their lifetime is exceeded.

Needs are associated with a needIndicator for every *ConsumerAgentGroup* through the need indicator function  $ni : CAG \times N \rightarrow \mathbb{R}_{\geq 0}$ . Numerical values  $ni(cag, n)$  are associated with need  $n$  and *ConsumerAgentGroup*  $cag$ , within the  $NDS_{cag}$ , specifying the needs interpretation.

### 3.3.4 Communication modeling

A central aspect in communication is the influence on product perceptions of consumer agents. In addition to perception, communication modeling is depicted in Figure 4.

In IRPact, perceptions are shaped primarily through reported experiences of communicating adopters and word-of-mouth of other consumers. With advertisement messages by company agents as part of communication, it furthermore addresses component requirement CR 25 (allow change of the perception or preference of an agent by corporate agents).

Socially-based perception (CR 18) and behavior and belief influence is achieved by the exchange of consumer-originated perception manipulation messages  $COPM \ni copm = (c, \hat{c}, pa, p)$  through communication within the social network, providing adaptable communication channels (CR 19). These messages are based on their own perception. They change the (perceived) product attribute values of product attribute  $pa$  of product  $p$  of the receiving agent  $\hat{c}$ , depending on the perceived product values of the sender  $c$  of the message. In addition to the messages sent by consumer agents, company agents can

also send (product perception manipulation) messages as specified through the **advertisement scheme**.

If the receiver is not aware of the product yet, they become aware of it through receiving a message. Furthermore, a new perception of  $pa$  is added. This perception is based on the perception  $\mathbf{ppavm}(c, pa, t)$  of the sender  $c$  and the weight of the edge connecting them in the social graph (i.e.  $\mathbf{w}((r, o, m), t)$ , with  $r = \mathbf{csgam}(c), o = \mathbf{csgam}(\hat{c})$ ). How this perception is incorporated into the cognitive context of the receiving agent  $\hat{c}$  depends on their perception scheme  $ps$  (as parameterized through the  $\mathbf{ppsm}_{cag}$ , with  $cag$  being the respective consumer agent group of  $\hat{c}$ ). This allows for the integration of other model aspects through the different perception schemes and addresses the influence of communication on behavior or believes (CR 20).

The timing, frequency and choice of the recipient within the social network of messages are governed by the CommunicationScheme  $CS_{cag}$  of the respective consumer groups. This is done through communication events that connect messages with their temporal evaluation through the **process model**. Formally, this is described by the communication event mapping  $\mathbf{cem}_{cag} : COPM_{cag} \times T \rightarrow CE_{cag}$ , encompassed by the communication scheme of the different **consumer groups**  $cag$ . This provides heterogeneous and flexible communication schemes based on the social and spatial factors used for grouping the agents (meeting component requirement CR 21). A crucial component of these are the message activity distributions  $\mathcal{MAD}_{cag}$ , which characterize the number of messages per time unit the groups consumer agents send to connected consumer agents  $\hat{c}$ .

Messages creation is governed by message schemes, depending on the sender of the message and the state of the simulation, formalized as  $\mathbf{ms}_{cag} : C \times \mathfrak{M} \times T \rightarrow \mathcal{P}(COPM)$ , describing which messages consumer agents  $c \in C$  with  $\mathbf{cagm}(c) = cag$  send at time  $t \in T$  depending on the model state  $\mathfrak{M}$ . Together, these schemes form the communication scheme  $CS_{cag} = (\mathbf{ms}_{cag}, \mathbf{cem}_{cag})$ .

### Information modeling

Information is characterized through being proactively sought by agents. Information seeking is governed by schemes invoked by the process model or the respective decision processes, describing what information agents think they require and what strategies they use to find it. Information in IRPact is understood in a naive form, as a piece of knowledge  $i \in I$ , with  $I$  being the set of all information, originating from an **information agent**. Forms of handling information are specified through information schemes ( $is \in IS$  as pair  $is = (\mathbf{iavm}, \mathbf{pim})$ ), which define how information model mechanics work and thus describe the information ecosystem within the simulation through their availability and product information seeking behavior.

Information is processed by an agent according to the **perception scheme** by acting as a perception respective to the product attribute (specified by the information product attribute mapping function  $\mathbf{ipam} : I \rightarrow PA$ , defined for all product attribute information  $pai \in PAI \subset I$ ).

Information schemes  $IS$  specify what information is available for agents to process via the information availability map  $\text{iavm} : I \times T \rightarrow \mathbb{B}$ . They indicate whether a given information is 'known' at a given time  $t \in T$ , and how product information is sought by consumer agents through the product information map  $\text{pim} : C \times PA \times T \rightarrow \mathcal{P}(I)$ . It thus describes what information about a product attribute  $pa \in PA$  is relevant to a consumer agent  $c \in C$  at time  $t \in T$ .

### 3.4 Agent modeling

Agents in IRPact represent cognitive entities in the modeled context. These can be:

- *units of adoption* (in the case of **Consumer agents**),
- *companies or company employees* responsible for sales / distribution policies of producers (in the case of **Company agents**),
- *shops or distribution departments* (in the case of **POS agents**),
- *actors within the policy sphere* (as in the case of **Policy agents**).

As agents in the model are intended to exhibit cognitive processes, information plays an important role. Its interpretation in a social context depends on the credibility of the information source, modeled as informational authority  $ia \in [0, 1]$  innate to all agents, quantifying their informational credibility through a numerical value<sup>7</sup>. This value is used in processes involving information  $I$  and its interpretation depends on the modeled cognition of the respective agents.

#### 3.4.1 Consumer agents

Consumer agents describe cognitive entities that represent households or individual consumers, and are thus often the most important type of agents for the diffusion and adoption of technology. Their primary role in the model is to adopt products and to interact with other agents. Consumer agents are organized in groups, bundling common characteristics of the consumer agents and implementing consumer agent heterogeneity, elaborated on **below**. Consumer agents are characterized primarily through mental structures and their environment; something unique to them, however, are their individual *consumer attributes*  $ca_{c,t}$ , addressing component requirement CR 24.

A consumer agent  $c \in C$  (with  $C$  being the set of consumer agents) is formalized as a 8-tuple, whose state at time  $t$  is described as

---

<sup>7</sup> Although a universal credibility of an agent's information is unrealistic, and agent-inherent factors, their context, the nature of the information and their informational history are relevant for assessing the credibility of another agent, this would make the model a lot more complex. Since there is already little justification for informational aspects of product adoption from the analyzed innovation diffusion models, a simple approach was chosen in IRPact. More complicated informational mechanics can easily be extended or implemented using other model mechanisms.

$c_t = (cag_c, ca_{c,t}, loc_{c,t}, Pr_{c,t}, Paw_{c,t}, PAP_{c,t}, ap_{c,t}, csgam_c)$ , with

$cag_c$	Consumer Agent Group association indicates which <b>consumer agent group</b> $c$ is a member of $cag_c = \mathbf{cagm}(c)$ (with consumer agent group mapping $\mathbf{cagm} : C \rightarrow CAG$ associating consumer agent $c \in C$ with their respective group $cag \in CAG$ ).
$ca_{c,t}$	State of the consumer attributes $ca_{c,t}$ of $c$ at time $t$ , with the consumer attributes drawn from the distributions $\mathcal{CGA}_{cag}$ described <b>below</b> .
$loc_{c,t}$	Coordinates of $c$ within the <b>spatial model</b> $\mathfrak{S}$ , initialized by drawing from the spatial distribution $\mathcal{SD}_{cag_c}$ of the corresponding agent group $cag_c$ .
$Pr_{c,t}$	Preference vector describing the agents' <b>preferences</b> $pr_v^{c,t}$ for value $v$ at time $t$ as entries ( $pr_v^{c,t} = \mathbf{pm}_c(v, t)$ ).
$Paw_{c,t}$	Product awareness vector for time $t$ and agents $c$ , indicating what products $p \in P$ agent $c$ is aware of.
$PAP_{c,t}$	Product attribute perception vector, indicating which perception $c$ has of the value of different product attributes at time $t$ as a numerical value.
$ap_{c,t}$	Set of adopted products $ap_{c,t} = \mathbf{pam}(c, t)$ , indicating which products are adopted by $c \in C$ at time $t \in T$ .
$csgam_c$	Consumer agent social graph association $csgam_c = \mathbf{csgam}(c)$ , describing which node $r$ in the <b>Social Graph</b> corresponds to $c$ , with consumer social graph association mapping $\mathbf{csgam} : C \rightarrow AN$ .

### 3.4.2 Consumer agent groups

To achieve the flexibility and balance between homogeneity and heterogeneity required by CR 25 (intra-homogenous and inter-heterogeneous agent groups), IRPact allows for flexible consumer agent groups ( $cag \in CAG$ ) which describe types of consumer agents through abstractions of the consumers. In addition to parameters linked to model dynamics, a  $cag$  specifies the distribution on which concrete values of its members are based upon. ConsumerAgentGroups thus serve as a template or blue print for consumer agents that play a decisive role in consumer agent instantiation, where its (numerical) values are drawn from the corresponding distribution, allowing for fine control over the homogeneity and heterogeneity of agents grouped together. A  $cag$  is formalized as an 11-tuple:

$$cag = (\mathcal{CGA}_{cag}, \mathcal{SD}_{cag}, \mathcal{CPD}_{cag}, \mathcal{PAWD}_{cag}, \mathbf{ppsm}_{cag}, d, \\ NDS_{cag}, CS_{cag}, \mathcal{IFPAD}_{cag}, \mathcal{FPAD}_{cag}, ia_{cag}), \text{ with}$$

$\mathcal{CGA}_{cag}$	Consumer group attribute vector containing the distributions the corresponding consumer agent attributes $ca_c$ of members of the agent group are drawn from.
$\mathcal{SD}_{cag}$	Spatial distribution giving the coordinates of agents of $cag$ within the <b>spatial model</b> $\mathfrak{S}$ .
$\mathcal{CPD}_{cag}$	<b>Preference</b> distribution from which initial groups agent preferences are drawn from.
$\mathcal{PAWD}_{cag}$	Set of product <b>awareness</b> distributions the initial awareness of corresponding agents are based on.
$\text{ppsm}_{cag}$	Product perception scheme mapping $\text{ppsm}_{cag} : PGA \rightarrow PS$ , assigning product group attributes $pga \in PGA$ their perception scheme $ps \in PS$ for <b>perceptions</b> $PAP_{c,t}$ of derived consumers.
$d$	Decision process that agents of $cag$ employ.
$NDS_{cag}$	<b>Need development scheme</b> for $cag$ .
$CS_{cag}$	<b>Communication scheme</b> $CS_{cag} = (\text{ms}_{cag}, \text{cem}_{cag})$ , specifying how messages and communication events are created through the message scheme $\text{ms}_{cag}$ and the communication event scheme $\text{cem}_{cag}$ .
$\mathcal{IFPAD}_{cag}$	Set of initial <b>fixed product</b> adoption distribution of $cag$ for the respective <b>fixed products</b> , describing their dissemination within $cag$ at $t = 0$ .
$\mathcal{FPAD}_{cag}$	Fixed product awareness distribution set of $cag$ which details product awareness instantiation for fixed products $fp$ at $t = 0$ .
$ia_{cag}$	<b>information authority</b> of agents of $cag$ .

### 3.4.3 Company agents

To allow for the required versatility, company agents must provide product-relevant consumer influence (CR 26) and active management and operation capabilities (CR 27). Active management and operation of a company agent  $coa \in COA$  in IRPact translates to making management decisions, as well as the execution of advertisement and marketing. Decision processes are formalized through the management decision scheme  $MDS_{coa}$  and product management is carried out the product quality manipulation schemes  $PQMS_{coa}$  as manipulation of product perceptions through unidirectional messages. These concepts address component requirement CR 22 and 23 through product-relevant consumer influence. Product management manifests in the manipulation of the product portfolio in the simulation, addressing requirement CR 8 through the (exogeneously supplied) market introduction of products and discontinuation of products of **fixed products**  $fp$ .

While market introduction events activate products (allowing them for adoption, i.e.  $\text{apm}(p, t) = \text{true} \forall t \geq \hat{t}$ ), product discontinuation events make

a fixed product unavailable to customers (i.e.  $\mathbf{apm}(p, t) = \text{false} \forall t \geq \hat{t}$ ) for an event at scheduled time  $\hat{t}$ .

The state of a company agent at time  $t$  is described as:

$$coa_t = (PP_{coa,t}, PQMS_{coa}, MDS_{coa}, AS_{coa}, ia_{coa}), \text{ with}$$

$PP_{coa,t}$	Product portfolio of $coa$ at time $t$ , describing which products the company agent manages $PP_{coa,t} \subset P_t$ .
$PQMS_{coa}$	Product quality manipulation scheme, representing how the product attributes of the products in the product portfolio of the $coa$ can be manipulated.
$MDS_{coa}$	Management decision scheme $coa$ uses. Describes how the states of products in the product portfolio are managed and how these decisions are taken by the agent.
$AS_{coa}$	Advertisement scheme, formalizing how advertisement messages for products in their portfolio $PP_{coa,t}$ are sent.
$ia_{coa}$	Information authority as an informational agent.

#### 3.4.4 Sales agents

As mentioned in 2.4, a POS needs to exhibit a product portfolio (CR 28), provide a purchase process, allow for spatial contextualization and be able to actively manage and operate their business (CR 29). In order to address these component requirements, a POS is parameterized through their dynamic behavior, location in space (if applicable) and a portfolio of available products with a POS-specific prize, allowing to investigate supply limitations and roll-out strategies.

Formally, a point-of-sale  $pos \in POS$  (with  $POS$  being the set of points-of-sale within the model) at simulation time  $t$  is described as a 5-tuple

$$pos_t = (Av_{pos,t}, PPr_{pos,t}, loc_{pos}, ia_{pos}, PuPS_{pos}), \text{ with}$$

$Av_{pos,t}$	Product availability vector $Av_{pos,t} = (av_p^{pos,t})_{p \in P} \in \mathbb{B}^{ P }$ , indicating whether the respective products $p$ are available at $pos$ at the time $t$ .
$PPr_{pos,t}$	Price vector for all products the $pos$ holds ( $av_p^{pos,t} = true$ ) at time $t$ for products $p \in P$ , relative to a reference price.
$loc_{pos}$	Placement in the spatial model $\mathfrak{S}$ (as coordinates), analogous to <b>consumer agents</b> , with the exception of being static.
$ia_{pos}$	Information authority as an informational agent.
$PuPS_{pos}$	Purchase process scheme modeling the purchase process at this POS.

### 3.4.5 Policy agent

Agents within the public sphere contain both exogenous and endogenous aspects. In IRPact, both perspectives are provided by a singular policy agent who manages existing policies and introduces new ones, bundling regulatory aspects falling in the public sphere. This includes both reactive and proactive policy management, addressing CR 30. Policies fall into three kinds of categories, namely product-directed policies ( $PPS_{poa}$ ), prohibitive policies ( $RPS_{poa}$ ) and consumer-directed policies ( $CPS_{poa}$ ), which are formalized through static (exogenous) or dynamic (endogenous) schemes. In order to be reactive (and meaningfully endogenous), market evaluation should take place as well. The policy agent  $poa$  itself is a static entity (independent of  $t$  in its state), and is formalized as follows:

$$poa = (PPS_{poa}, CPS_{poa}, RPS_{poa}, MES_{poa}, ia_{poa}), \text{ with}$$

$PPS_{poa}$	Product-directed policy scheme, describing when product attributes of what products are manipulated.
$CPS_{poa}$	Consumer-directed policy scheme, specifying what policies are used by the policy agent to influence perceptions and preferences of consumer agents.
$RPS_{poa}$	Regulatory policy scheme, formalizing restrictions through discontinuation of products.
$MES_{poa}$	Market evaluation scheme, describing how the policy agent derives information about market and agents.
$ia_{poa}$	Informational authority of the policy agent $poa$ as an information agent.

## 4 Software implementation

Based on the model design in the previous section, a software implementation was written in Java 1.8. The publicly available source code was published on github<sup>8</sup> and is available under the GPLv3 license. A javadoc-based documentation and a structured discussion can be found as well. In addition, the model is archived on the CoMSES Computational Model Library<sup>9</sup>.

The code base is designed to be highly modular and object-centered. Group-instance relationships are realized via attributes and are modeled as instances-partOf relations. The factory method pattern was widely used to create most entities like agents and products. In addition, most of the behavior of the entities is handled by interfaces, called schemes. This labeling is based on the schemes of the model design. Both approaches allow individual implementations to be easily interchangeable. In IRPact tasks or interactions between

<sup>8</sup> <https://github.com/Simonjohanning/IRPact>

<sup>9</sup> <https://www.comses.net/codebase-release/04a946bf-10e3-4700-9128-0d42e18663d7/>



agents like communication are implemented as events, which can be scheduled and executed at a specific point in time.

The component diagram in Figure 6 visualizes the structure of IRPact. It shows the modular aspect based on subsystems. These are primarily grounded on the modelling approaches. For a better understanding and distinction, schema components were labeled as schema.

In the following, the concrete implementation of the presented framework model is described. Due to reasons of space and for a better overview, only key classes are considered. The full structure can be found in the mentioned repository.

## 4.1 Fundamentals implementation

### 4.1.1 Temporal model

The temporal model describes the temporal dynamics of the simulation. It is implemented as an abstract `TimeModel` class to support different temporal modes. The primary task is to govern the temporal order. This aspect also includes starting and controlling the whole simulation. By monitoring the temporal order the `TimeModel` also holds a reference to the `EventScheduler`, which handles the scheduling and execution of events. In addition it contains a reference to the `ProcessModel` to trigger adoption processes.

### 4.1.2 Process model

The abstract `ProcessModel` governs the execution of dynamic aspects of the model. In general the process model specifies the order of execution of processes. The concrete realization is based on the used time model. For this reason the implementation of the process model strongly depends on the time model. To specify how consumer agents act when ceasing to adopt a product, the `ProcessModel` utilizes an `AdoptionReplacementScheme`. This scheme describes how adopted products are replaced once they are discontinued on the market, and how consumer agents respond to these discontinuations.

### 4.1.3 Spatial model

The spatial model of the simulation is realized using the abstract `SpatialModel` class. It defines how spatial entities are situated to each other based on a metric scheme. In addition this class also implements the spatial border map `sbm` to decide if an entity is within bounds.

### 4.1.4 Social network

The social network *SN* was implemented closely to the modelling approach. While the `EdgeWeightManipulationScheme` interface implements the edge

weight function `w` and the edge weight scheme `ews`, the `TopologyManipulationScheme` interface implements the topology mutability scheme `tms`. Both enable changes to the underlying social graph, which is implemented in the `SocialGraph` class. All three parts are combined in the `SocialNetwork` class, which realizes the social network  $SN$  itself. As in modelling, consumer agents are nodes of the graph and edges response to interactions. The consumer agent social association mapping `csgam` was realized by the `ConsumerAgent` having a reference to the corresponding node in the graph.

## 4.2 Product implementation

The product implementation is heavily based on the modelling approach. `Product`, `ProductAttribute`, `ProductGroup` and `ProductGroupAttribute` classes implementing the functionality in accordance to Section 3.2. To realize the product group association map `pgam`, the `Product` class holds a reference to their respective `ProductGroup` instance.

Fixed products are also realized by the `Product` class. The fixed product awareness distributions  $\mathcal{FPA}$  and the initial fixed product adoption distributions  $\mathcal{IFPA}$  are part of the `ConsumerAgentGroup` class. Both are implemented as simple maps.

## 4.3 Mental structures implementation

### 4.3.1 Perception modeling

Product perception is realised through two different concepts. The `ProductAttributePerceptionScheme` interface calculates the actual perception of a product attribute for a respective agent at a specific point of time. For that reason the `ConsumerAgent` class contains a perception scheme for every product attribute. This mapping corresponds to the perceived product attribute value map `ppavm`. The product perception scheme mapping `ppsmcag` is realised through the `PerceptionSchemeConfiguration` class, which is part of the `ConsumerAgentGroup`.

Because product observation or post-purchase evaluation is an active task, they are implemented as events. The `PostPurchaseEvaluationEvent` handles the addition of product perception to concerned consumer agents.

In order for consumer agents to know about the existence of products, the `ConsumerAgent` class manages a set of known products. If a product is part of the set, the corresponding agent is aware of it. Based on this set, the product awareness map `patvm` was realized.

### 4.3.2 Preference modeling

In order to realize the preference model, first the abstraction of values  $V$  was required. Since these are usually of descriptive nature to express the

state of mind of an agent, they are realized as strings in the simulation. A simple `Value` class holds the information. Combined with a numerical preference strength, a value is coupled in a `Preference` class, which realizes the preference mapping `pm`. In order to assign a preference to a specific information or object, a mapping between the value and the corresponding object is required. Currently only product attributes are considered as assignable targets to evaluate the (agent specific) utility in decision processes. The `ProductGroupAttributeValueMapping` class maps a product group attribute with the corresponding value based on a mapping strength. This implementation corresponds to the product attribute value preference mapping `paupm`. The value instance in both, the `Preference` class and the `ProductGroupAttributeValueMapping` class, can be used to obtain the preference strength of the corresponding product attribute.

#### 4.3.3 Decision modeling

Decisions can be of various kinds. The marker interface `DecisionMakingProcess` forms the base for all decision processes. The `ConsumerAgentAdoptionDecisionProcess` interface realizes the extended product adoption decision map `cpadm` (and thus the normal `padm`). It offers methods to select a product to adopt and to find superior alternatives, if available.

#### 4.3.4 Needs modeling

As mentioned in Section 3.3.3, the need concept (and need events) are closely connected to decision processes. Needs, like values, are an abstraction based on the same data type. In IRPact both are based on strings. Needs gain their semantics through other model components, such as the abstract `NeedDevelopmentScheme` class. This scheme describes how needs develop and what happens when a product expires. The latter is important in order to decide whether already satisfied needs should be active again.

In order to work and react to needs, they are embedded in an `NeedEvent` class. That approach allows them to be processed at a specific point in time.

Figure 7 shows all actions taking place to satisfy a need. Because the `ProcessModel` controls the model dynamics, it is responsible for invoking the `NeedDevelopmentScheme`. Based on the concerned consumer agent, the scheme creates all necessary `NeedEvents`. Like other events, these are scheduled on the `EventScheduler`. By executing a `NeedEvent` the adoption process is started. First, all products that satisfy the corresponding need are collected. Based on the `ConsumerAgentAdoptionDecisionProcess` of the concerned consumer agent, the most suitable product is selected. Finally, the selected product is adopted and subsequent processes are invoked.

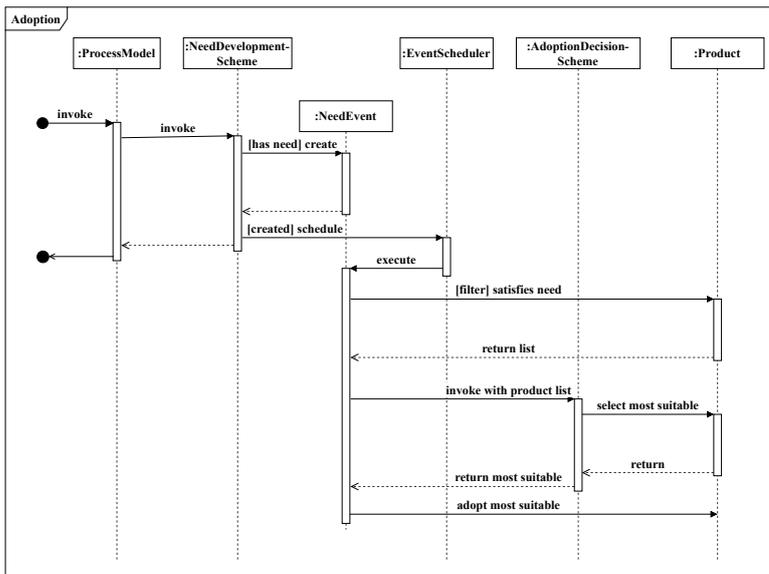


Figure 7 Visualization of the workflow to satisfy a need through product adaptation.

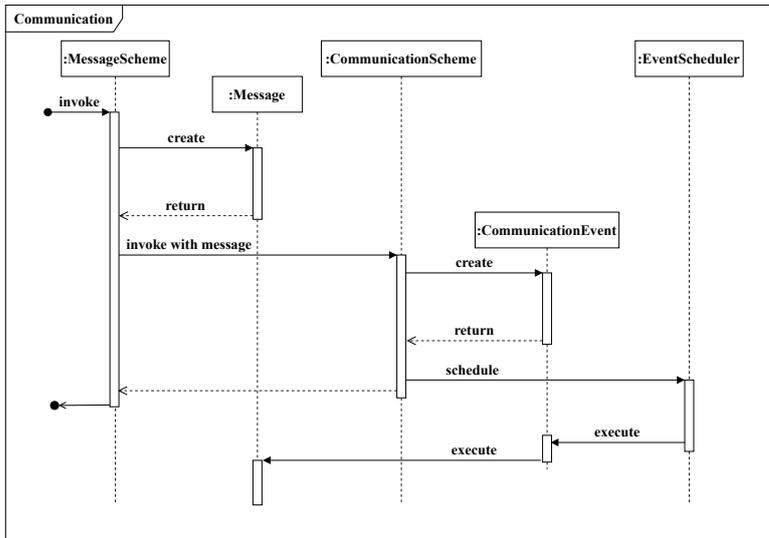
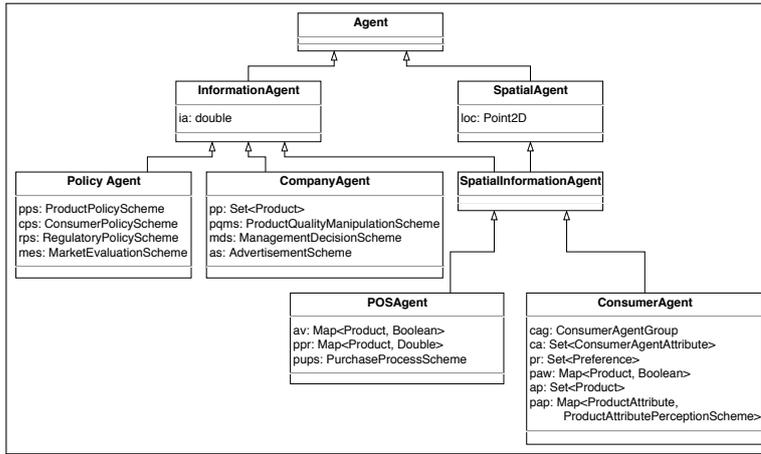


Figure 8 Interactions between MessageScheme, CommunicationScheme and EventScheduler when handling messages.

#### 4.3.5 Communication modeling

Communication between agents is based on messages. In IRPact messages are instructions. Figure 8 shows the full communication process and all involved components. The MessageScheme interface handles the creation of Message



**Figure 9** Class diagram showing the IRPact agent hierarchy including all relevant fields. Field names are based on parameter names used in Section 3.4.

instances for a sending agent. Based on the scheme the receiving agent and the instructions are specified. Next, the created set of messages is processed by a **CommunicationScheme** to create a set of **CommunicationEvents**, realizing the communication event mapping **cem**. A communication event links a message to the temporal model and allows to schedule instructions and events. Finally the created communication events are scheduled by the **EventScheduler** and executed on their due date.

#### 4.3.6 Information modeling

As information is too abstract as a concept to have any meaningful implementation in IRPact, *Information* was chosen to be an abstract class, with only holds a reference to the agent from which it was provided. Information about product qualities is modeled through the *ProductAttributeInformation* and corresponds to *PAI*. The way information enters the simulation and is processed is implemented by *InformationSchemes*.

#### 4.4 Agent modeling

In IRPact all agents are extensions of the **Agent** class. Figure 9 shows the concrete hierarchy. Based on their functionality agents are divided into two categories. **SpatialAgents** possess spatial information, implemented as a (2-dimensional) coordinate, to model the location *loc*. **InformationAgents** have the ability to share information. The information authority *ia* is modelled using the *ia* attribute of the **InformationAgents**. **SpatialInformationAgent** combines both extensions.

Because all concrete modeled agents in Section 3.4 are required to have the ability to share information, their implementations are all extensions of the `InformationAgent` class. In addition, consumer agents and POS agents require an additional location *loc*. Therefore they are extensions of the `SpatialInformationAgent` class. The specific behaviour of all agents is mediated through schemes. Figure 9 shows the kind of implementation of the parameters of each agent type. In contrast, the subsystem `Agents` in the component diagram in Figure 6 displays the composition of each agent, their used schemes and the relations to other components.

## 5 Conclusion

This article started out from the business need of assessing the diffusion of new innovations in a dynamic market. Despite the popularity of eABMs, no common agent-based framework for innovation diffusion exists. In order to provide a theoretically grounded infrastructure for these models, this research set out to develop a flexible, modular and extensible common description and implementation framework for existing and future models.

This was addressed through deriving the high-level requirements of flexibility and modularity (R2) and depicting the variety of existing model components. For the latter (R1), 30 more granular component requirements were derived based on an analysis of existing empirically-grounded agent-based innovation diffusion models.

The component requirements were addressed through the formal definition of the framework in four model layers: model foundations, products, mental structures and agents. Based on this, a software implementation allowing for easy configuration and execution of existing and future models within this framework was written in the Java programming language.

Both, the formal model and the software implementation, contain the large variety of components within existing models identified in this article. As both the model framework and the implementation are flexible, modular and extensible and address all component requirements, it can be concluded that the solution objective is met by the approach.

As a variable decision support tool in decision-making processes on the adoption of innovations that is built on modern existing approaches in the literature, we believe that IRPact is valuable for the investigation of a range of research questions on innovation diffusion, business model evaluation and infrastructure transformation.

Future work will focus primarily on the application of this framework. Serving as a proof of concept for the feasibility of the framework, a concrete model for the adoption of rooftop photovoltaic systems by consumers in an urban setting is currently under development. So far, the framework has shown to be a valuable tool for this model and we further expect the model to refine insights on several aspects, such as decision processes, products and the policy agent. The application of a concrete model within IRPact would further

prove the practical and scientific value of the framework. Through it, both the interrelations between customer behavior and incentive measures and practical evaluation of business model innovations or policy measures could be evaluated.

Through embedding the framework in a multi-model infrastructure as shown in (25), more comprehensive research questions on infrastructure transformation could furthermore be addressed. IRPact is developed as the socio-economic counterpart to the techno-economical infrastructure modeling tool IRPopt. Coupling these two perspectives on infrastructure transformation allows to investigate crucial research questions more in-depth and in a more sophisticated manner. In addition to proving its value as a socio-economic modeling tool, insights from this interrelated analysis would identify further development potential for IRPact itself.

## Declarations

### Availability of data and materials

The sourcecode of the framework is publically available on github (<https://github.com/Simonjohanning/IRPact>), and the model is archived on the CoMSES Computational Model Library (<https://www.comses.net/codebase-release/04a946bf-10e3-4700-9128-0d42e18663d7/>).

### Competing Interests

The authors declare that they have no competing interests.

### Funding sources

Simon Johanning, Fabian Scheller, Daniel Abitz, and Claudius Wehner receive funding from the project Smart Utilities and Sustainable Infrastructure Change (SUSIC) with the project number 1722 0710. This study is financed by the Saxon State government out of the State budget approved by the Saxon State Parliament. Fabian Scheller also receives funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement no. 713683 (COFUNDfellowsDTU).

### Authors' contribution

Simon Johanning and Fabian Scheller developed the conception and the design of the framework and were chiefly responsible for writing the article (with

Simon Johanning being the principal author and developer). Model implementation was done primarily by Simon Johanning, with Daniel Abitz contributing to it. Daniel Abitz and Claudius Wehner provided substantial textual input and created the graphical representation of the framework. Thomas Bruckner provided feedback and guidance and participated in discussions regarding the content and the manuscript of the framework.

#### Acknowledgement

We gratefully thank the peer-reviewers of this article for their feedback and the editor of the journal. Furthermore, we want to thank Prof. Middendorf for his feedback on the framework during the draft.

#### References

1. S. Balbi, C. Giupponi, P. Perez, and M. Alberti. A spatial agent-based model for assessing strategies of adaptation to climate and tourism demand changes in an alpine tourism destination. *Environmental Modelling & Software*, 45:29–51, 2013. ISSN 1364-8152. doi: 10.1016/j.envsoft.2012.10.004.
2. O. Barreteau, E. Sauquet, J. Riaux, N. Gailliard, and R. Barbier. Agent based simulation of drought management in practice. In *Advances in social simulation*, pages 237–248. Springer, 2014. ISBN 3642398286. doi: 10.1007/978-3-642-39829-2\_21.
3. F. M. Bass. A new product growth for model consumer durables. *Management Science*, 15(5):215–227, 1969. ISSN 0025-1909. doi: 10.1287/mnsc.15.5.215.
4. A. R. Bell, D. T. Robinson, A. Malik, and S. Dewal. Modular abm development for improved dissemination and training. *Environmental Modelling & Software*, 73:189–200, 2015. ISSN 1364-8152. doi: 10.1016/j.envsoft.2015.07.016.
5. R. Boero and F. Squazzoni. Does empirical embeddedness matter? Methodological issues on agent-based models for analytical social science. *Journal of Artificial Societies and Social Simulation*, 8(4), 2005.
6. E. Bonabeau. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences of the United States of America*, 99(Suppl 3):7280–7287, 2002. ISSN 0027-8424. doi: 10.1073/pnas.082080899.
7. T. L. J. Broekhuizen, S. A. Delre, and A. Torres. Simulating the Cinema Market: How Cross-Cultural Differences in Social Influence Explain Box Office Distributions. *Journal of Product Innovation Management*, 28(2): 204–217, 2011. ISSN 1540-5885. doi: 10.1111/j.1540-5885.2011.00792.x.
8. C. M. Buchmann, K. Grossmann, and N. Schwarz. How agent heterogeneity, model structure and input data determine the performance of

- an empirical ABM – A real-world case study on residential mobility. Environmental Modelling & Software, 75:77–93, 2016. ISSN 1364-8152. doi: 10.1016/j.envsoft.2015.10.005.
9. E. J. Chappin and M. R. Afman. An agent-based model of transitions in consumer lighting: Policy impacts from the E.U. phase-out of incandescents. Environmental Innovation and Societal Transitions, 7:16–36, 2013. ISSN 2210-4224. doi: 10.1016/j.eist.2012.11.005.
  10. S. A. Delre, W. Jager, T. H. A. Bijmolt, and M. A. Janssen. Will It Spread or Not? The Effects of Social Influences and Network Topology on Innovation Diffusion. Journal of Product Innovation Management, 27(2): 267–282, 2010. ISSN 1540-5885. doi: 10.1111/j.1540-5885.2010.00714.x.
  11. A. G. Dunn and B. Gallego. Diffusion of Competing Innovations: The Effects of Network Structure on the Provision of Healthcare. Journal of Artificial Societies and Social Simulation, 13(4), 2010. ISSN 1460-7425. doi: 10.18564/jasss.1690.
  12. S. N. Durlauf. Complexity, economics, and public policy. Politics, Philosophy & Economics, 11(1):45–75, 2012. ISSN 1470-594X. doi: 10.1177/1470594X11434625.
  13. M. J. Eppstein, D. K. Grover, J. S. Marshall, and D. M. Rizzo. An agent-based model to study market penetration of plug-in hybrid electric vehicles. Energy Policy, 39(6):3789–3802, 2011. ISSN 0301-4215.
  14. E. R. Frederiks, K. Stenner, and E. V. Hobman. Household energy use: Applying behavioural economics to understand consumer decision-making and behaviour. Renewable and Sustainable Energy Reviews, 41:1385–1394, 2015. ISSN 13640321. doi: 10.1016/j.rser.2014.09.026.
  15. B. G. Glaser and A. L. Strauss. The discovery of grounded theory. Chicago (US): Aldine, 1967.
  16. M. Graebig, G. Erdmann, and S. Röder, editors. Assessment of residential battery systems (RBS): profitability, perceived value proposition, and potential business models, 2014.
  17. T. Jensen, G. Holtz, and É. J. Chappin. Agent-based assessment framework for behavior-changing feedback devices: Spreading of devices and heating behavior. Technological Forecasting and Social Change, 98:105–119, 2015. ISSN 00401625. doi: 10.1016/j.techfore.2015.06.006.
  18. E. Kiesling. Planning the market introduction of new products: An agent-based simulation of innovation diffusion. UBW Volltext am Hochschulschriftenserver der UB Wien, Wien, 2011. URL <http://othes.univie.ac.at/16743/>.
  19. E. Kiesling, M. Günther, C. Stummer, and L. M. Wakolbinger. Agent-based simulation of innovation diffusion: a review. Central European Journal of Operations Research, 20(2):183–230, 2012. ISSN 1435-246X. doi: 10.1007/s10100-011-0210-y.
  20. F. Kostadinov, S. Holm, B. Steubing, O. Thees, and R. Lemm. Simulation of a Swiss wood fuel and roundwood market: An explorative study in agent-based modeling. Forest Policy and Economics, 38:105–118, 2014. ISSN 13899341. doi: 10.1016/j.forpol.2013.08.001.

21. C. M. Macal and M. J. North. Tutorial on agent-based modelling and simulation. Journal of simulation, 4(3):151–162, 2010. ISSN 1747-7778. doi: 10.1057/jos.2010.3.
22. D. McCoy and S. Lyons. Consumer preferences and the influence of networks in electric vehicle diffusion: An agent-based microsimulation in Ireland. Energy Research & Social Science, 3:89–101, 2014. ISSN 22146296. doi: 10.1016/j.erss.2014.07.008.
23. J. Palmer, G. Sorda, and R. Madlener. Modeling the diffusion of residential photovoltaic systems in Italy: An agent-based simulation. Technological Forecasting and Social Change, 99:106–131, 2015. ISSN 00401625. doi: 10.1016/j.techfore.2015.06.011.
24. V. Rai and S. A. Robinson. Agent-based modeling of energy technology adoption: Empirical integration of social, behavioral, economic, and environmental factors. Environmental Modelling & Software, 70:163–177, 2015. ISSN 1364-8152. doi: 10.1016/j.envsoft.2015.04.014.
25. D. G. Reichelt, S. Kühne, F. Scheller, D. Abitz, and S. Johanning. Towards an Infrastructure for Energy Model Computation and Linkage. Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, 2020.
26. M. G. Rodriguez, D. Balduzzi, and B. Schölkopf. Uncovering the temporal dynamics of diffusion networks. arXiv preprint arXiv:1105.0697, 2011.
27. E. M. Rogers. Diffusion of innovations. Free Press, New York, 5th ed. edition, 2003. ISBN 9780743222099.
28. F. Scheller and T. Bruckner. Energy system optimization at the municipal level: An analysis of modeling approaches and challenges. Renewable and Sustainable Energy Reviews, 105:444–461, 2019. doi: 10.1016/j.rser.2019.02.005.
29. F. Scheller, S. Johanning, and T. Bruckner. A review of designing empirically grounded agent-based models of innovation diffusion: Development process, conceptual foundation and research agenda. Research Report No.01 (2019), Leipzig University, Institute for Infrastructure and Resources Management (IIRM), 2019. URL <http://hdl.handle.net/10419/191981>.
30. M. E. Schramm, K. J. Trainor, M. Shanker, and M. Y. Hu. An agent-based diffusion model with consumer and brand agents. Decision Support Systems, 50(1):234–242, 2010. ISSN 01679236. doi: 10.1016/j.dss.2010.08.004.
31. N. Schwarz. Umweltinnovationen und Lebensstile: Eine raumbezogene, empirisch fundierte Multi-Agenten-Simulation, volume 3. Metropolis-Verlag GmbH, 2007.
32. N. Schwarz and A. Ernst. Agent-based modeling of the diffusion of environmental innovations — An empirical approach. Technological Forecasting and Social Change, 76(4):497–511, 2009. ISSN 00401625. doi: 10.1016/j.techfore.2008.03.024.
33. A. Smajgl and O. Barreteau, editors. Empirical Agent-Based Modelling - Challenges and Solutions. Springer New York, New York, NY, 2014. ISBN 978-1-4614-6133-3. doi:

- 10.1007/978-1-4614-6134-0.
34. A. Smajgl and O. Barreteau. Series foreword. In A. Smajgl and O. Barreteau, editors, Empirical Agent-Based Modelling - Challenges and Solutions, pages v–viii. Springer New York, New York, NY, 2014. ISBN 978-1-4614-6133-3. doi: 10.1007/978-1-4614-6134-0.
  35. B. M. Sopha, C. A. Klöckner, and E. G. Hertwich. Adoption and diffusion of heating systems in Norway: coupling agent-based modeling with empirical research. Environmental Innovation and Societal Transitions, 8: 42–61, 2013. ISSN 2210-4224. doi: 10.1016/j.eist.2013.06.001.
  36. C. Stummer, E. Kiesling, M. Günther, and R. Vetschera. Innovation diffusion of repeat purchase products in a competitive market: An agent-based simulation approach. European Journal of Operational Research, 245(1): 157–167, 2015. ISSN 0377-2217. doi: 10.1016/j.ejor.2015.03.008.
  37. C. Swinerd and K. R. McNaught. Simulating the diffusion of technological innovation with an integrated hybrid agent-based system dynamics model. Journal of simulation, 8(3):231–240, 2014. ISSN 1747-7778. doi: 10.1057/jos.2014.2.
  38. P. S. van Eck, W. Jäger, and P. S. H. Leeflang. Opinion leaders’ role in innovation diffusion: A simulation study. Journal of Product Innovation Management, 28(2):187–203, 2011. ISSN 1540-5885. doi: 10.1111/j.1540-5885.2011.00791.x.
  39. P. Windrum, T. Ciarli, and C. Birchenhall. Consumer heterogeneity and the development of environmentally friendly technologies. Technological Forecasting and Social Change, 76(4):533–551, 2009. ISSN 00401625. doi: 10.1016/j.techfore.2008.04.011.
  40. I. Wolf, T. Schröder, J. Neumann, and G. de Haan. Changing minds about electric cars: An empirically grounded agent-based modeling approach. Technological Forecasting and Social Change, 94:269–285, 2015. ISSN 00401625. doi: 10.1016/j.techfore.2014.10.010.
  41. M. J. Wooldridge. Agent technology: foundations, applications, and markets. Springer Science & Business Media, 1998. ISBN 3540635912. doi: 10.1007/978-3-662-03678-5.
  42. T. Zhang and W. J. Nuttall. Evaluating Government’s Policies on Promoting Smart Metering Diffusion in Retail Electricity Markets via Agent-Based Simulation\*. Journal of Product Innovation Management, 28(2): 169–186, 2011. ISSN 1540-5885. doi: 10.1111/j.1540-5885.2011.00790.x.
  43. T. Zhang and W. J. Nuttall. Evaluating government’s policies on promoting smart metering diffusion in retail electricity markets via agent-based simulation\*. Journal of Product Innovation Management, 28(2):169–186, 2011. ISSN 1540-5885. doi: 10.1111/j.1540-5885.2011.00790.x.
  44. M. Zsifkovits. Agent-Based Modeling for Simulating Eco-Innovation Diffusion: A Review on the case of Green Mobility. International Journal of Science and Research (IJSR), 4(8):264–269, 2015.

# Figures

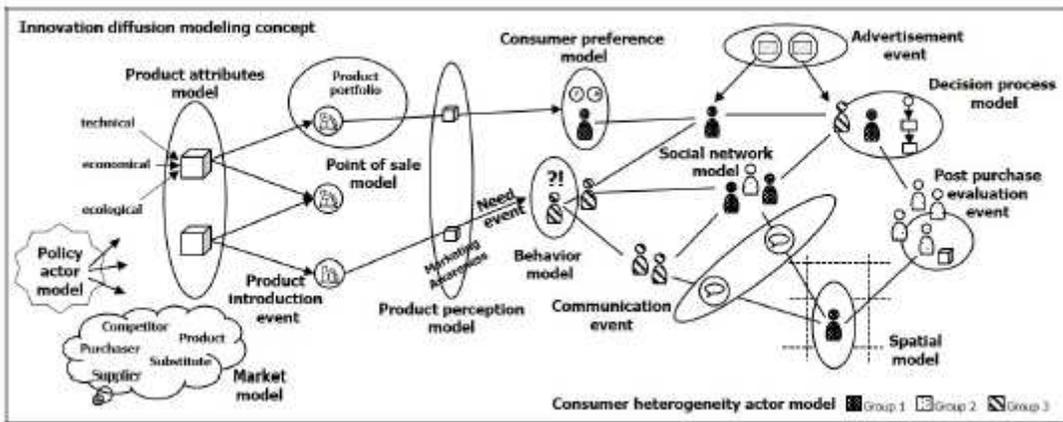


Figure 1

Illustrative representation of identified agent and product entities, model dynamics and simulation foundations (based on the results of (29))

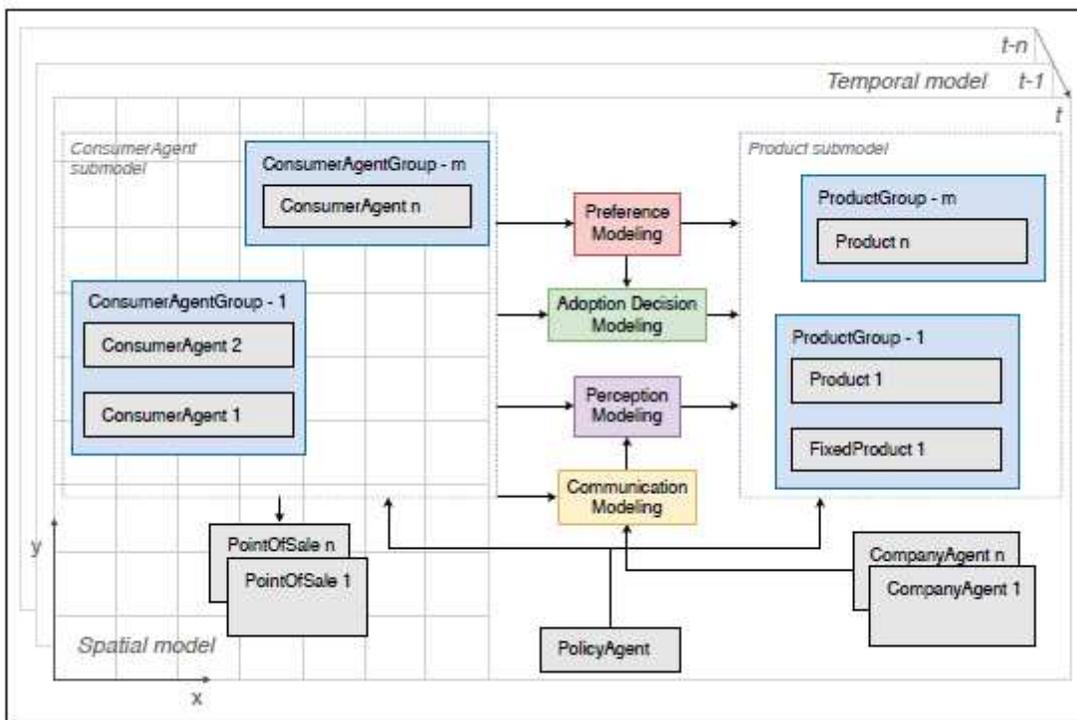


Figure 2

Overview of IRPact framework components.

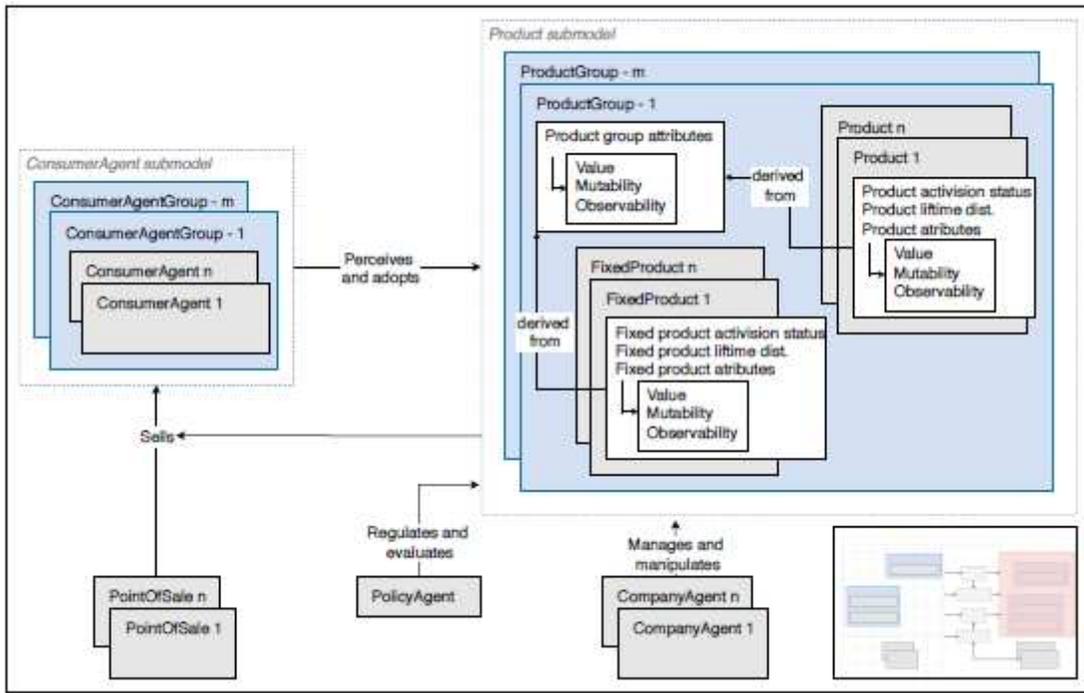


Figure 3

Product related entities of IRPact modeling components.

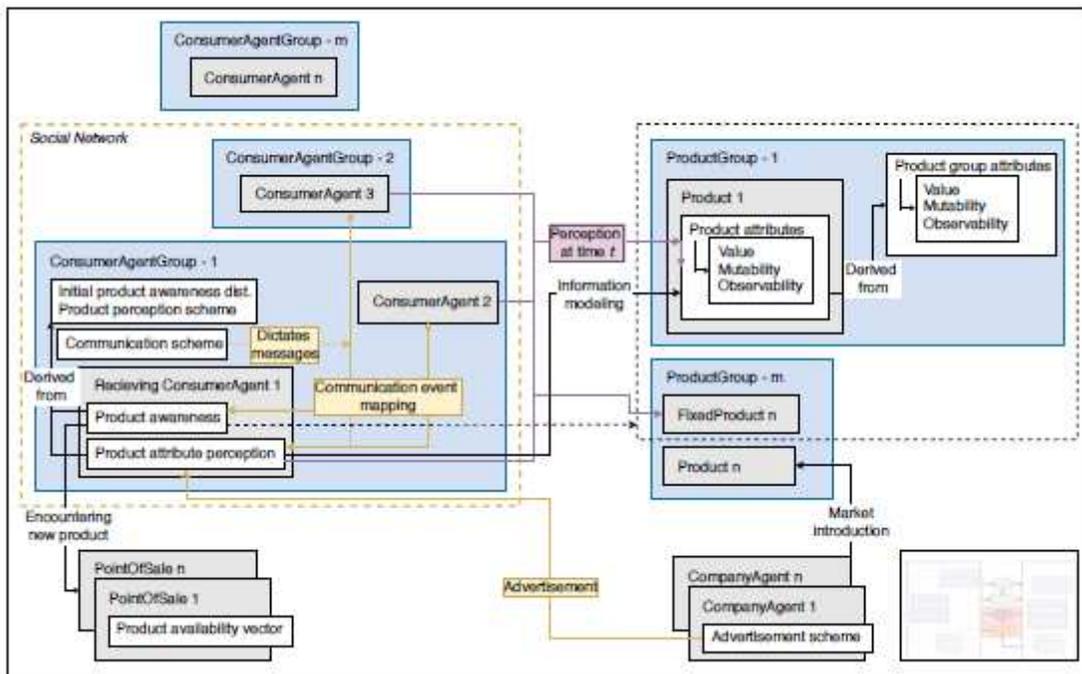
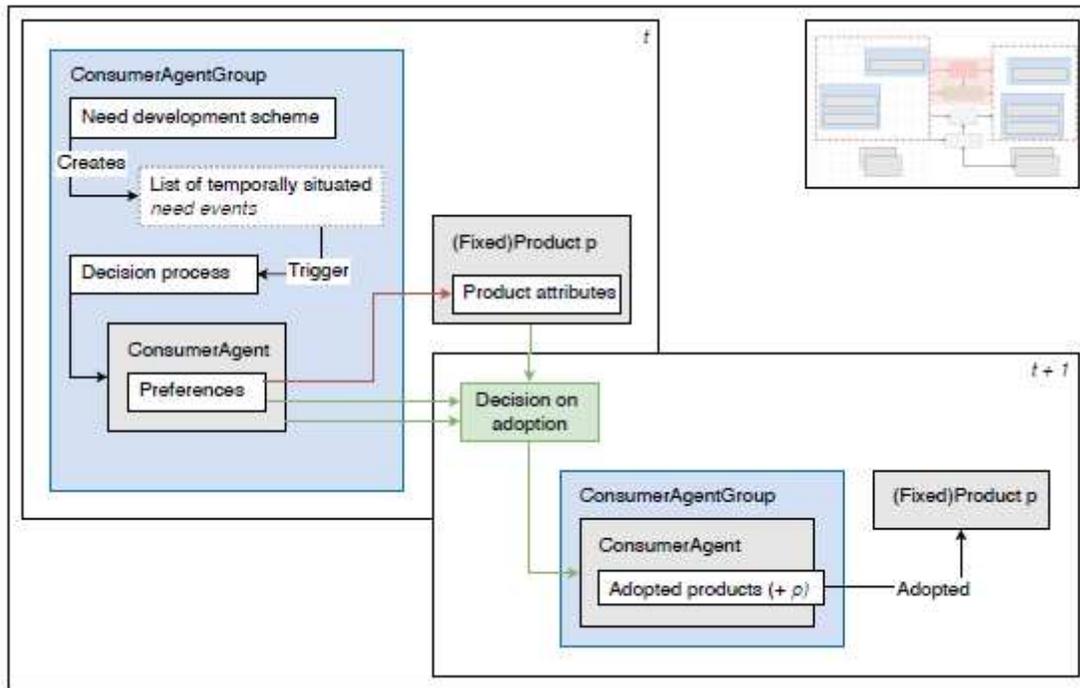


Figure 4

Perception, communication and information modelling with the relevant components of the IRPact framework.



**Figure 5**

Preference and adoption decision modeling process of a single consumer agent.



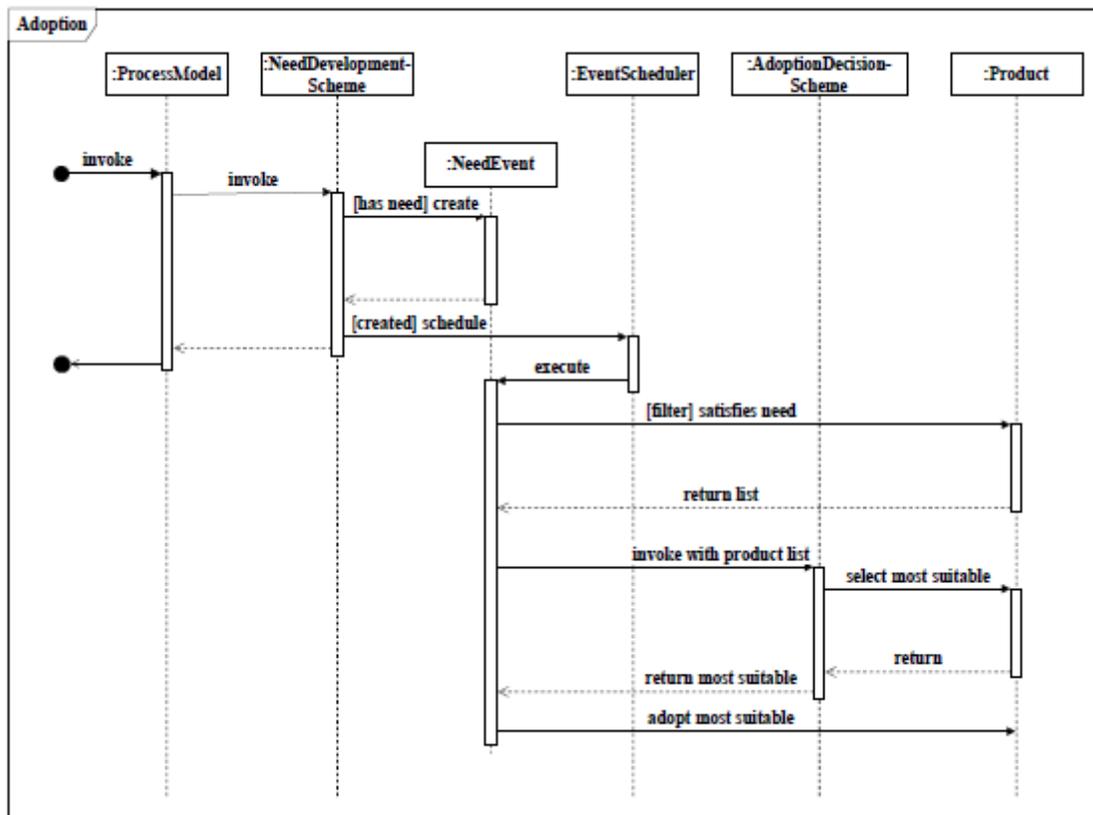


Figure 7

Visualization of the workflow to satisfy a need through product adaptation.

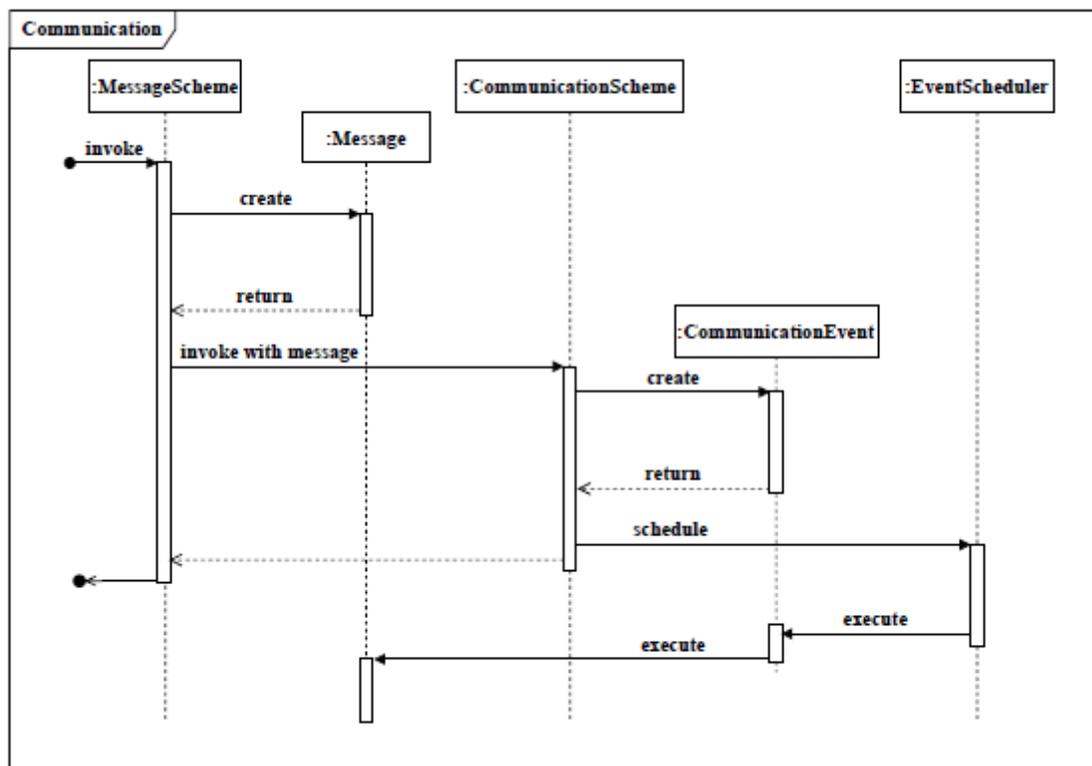


Figure 8

Interactions between MessageScheme, CommunicationScheme and EventScheduler when handling messages.

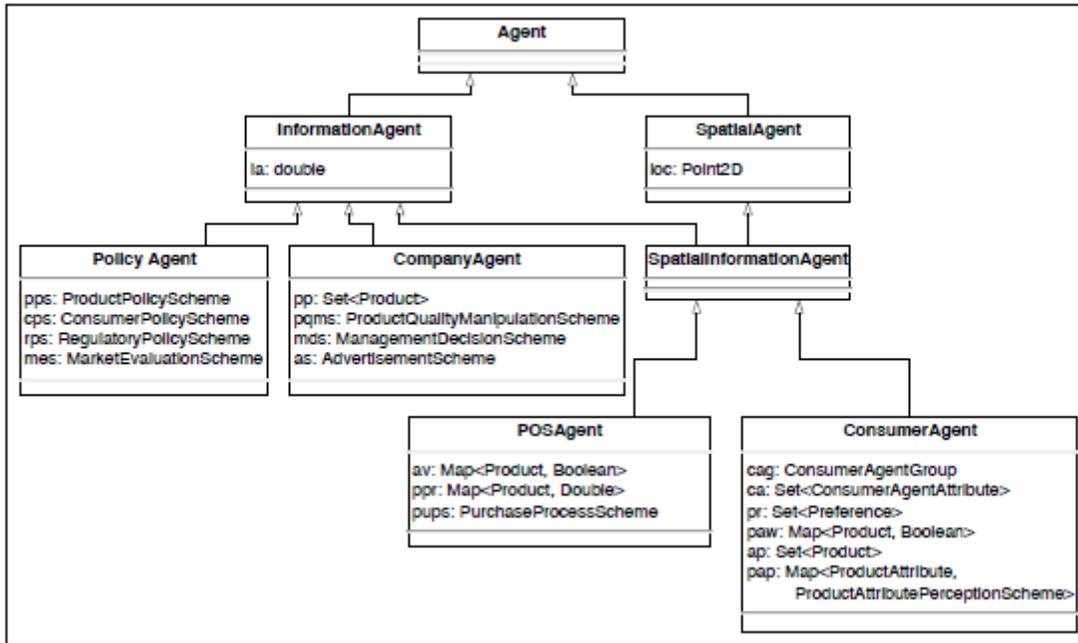


Figure 9

Class diagram showing the IRPact agent hierarchy including all relevant elds. Field names are based on parameter names used in Section 3.4.