



Evolutionary and Adaptive Elderly Care Ecosystem

Thais A. Baldissera^{1,2} and Luis M. Camarinha-Matos¹

¹ Faculty of Science and Technology and UNINOVA-CTS,
NOVA University of Lisbon, Campus de Caparica, 2829-516 Caparica, Portugal
{tab,cam}@uninova.pt

² Instituto Federal Farroupilha, Santa Maria, RS 97050-685, Brazil

Abstract. Personalized care services for each individual are typically required in the aging process. Considering this domain, a collaborative elderly care ecosystem (ECE) is proposed to support the provision of composite services that may combine contributions from multiple service providers. A personalized solution is built through of services composition, tailored to the individual customer (senior), respecting her/his requirements, preferences, lifestyle, and constraints. An additional issue the ecosystem must deal with is the problem of evolution, as individual's care needs are not static over time. Consequently, the care services need to evolve accordingly to keep the elderly's requirements satisfied. This process of services' adaptation is challenging since many services can be dependent on each other, and there are various constraints that need to be observed before adaptating and enacting new services. In this paper, we exploit socio-technical aspects of service adaptation in the context of elderly care ecosystems. Starting with a service personalization method previously proposed (SCoPE method), we presented an adaptive and evolutionary system (SEvol) based on the MAPE-K methodology. The SEvol method considers customer's inputs and suggests evolution plans. In the end, a workflow diagram is presented considering the main processes of ECE demonstrating the roles of the ECE environments and the main stakeholders, according to three stages: preparation, execution and monitoring phase.

Keywords: Elderly Care Ecosystem · Service Evolution ·
Service Composition and Personalization · Collaborative Networks

1 Introduction

Recent studies in the field of aging indicate that the seniors population worldwide is constantly growing [1–3]. In Europe, elderly population already represents 24% of the population (175 millions of persons), in contrast to 16% of youngsters (117 millions) [3]. While it is expected that the general Europe's populace decreases over the years, current trends indicates that the quantity of seniors will reach rates of 27.2% of the populace by 2050 [4].

In the well-being and aging context, a collaborative Elderly Care Ecosystem (ECE), in line with the notion of a Collaborative Business Ecosystem, “has the potential to provide an environment where personalized services might increase customer

satisfaction, and give service providers access to new opportunities, share costs and risks, and strengthen their business” [5]. In this context, Collaborative Networks are the cornerstone of collaboration initiatives among service providers. To accomplish these objectives, composition and rating of care and assistance services should meet particular needs, since customer care needs can be attended in distinct forms, considering various providers. This paper proposes a method to address this need.

The remainder of this article is organized as follows: the Elderly Care Ecosystem is introduced in Sect. 2. The Service Composition and Personalization Environment (SCoPE) method is also briefly mentioned. Section 3 introduces the ECE evolution system represented by the Service Evolution (SEvol) method. The ECE processes are described in Sect. 4. Finally, the conclusions are presented in Sect. 5.

2 Elderly Care Ecosystem - ECE

In general, a number of technologies has been employed to support organizations to work together better. The ubiquitous use of open distributed systems, such as internet-based systems, enables efficient distributed businesses process, faster time to market, and practical development through global economy [3]. In dynamic environments, such as found in health services, organizations are constantly confronted to efficiently collaborate with other enterprises and compose personalized offers without losing quality and competitiveness.

In this context, we can resort to the concept of collaborative network (CN) which is defined as “an alliance constituted of a variety of entities (e.g. organizations and people) that are largely autonomous, geographically distributed, and heterogeneous in terms of their operating environment, culture, social capital and goals, but that collaborate to better achieve common or compatible goals, and whose interactions are supported by a computer network” [6].

A particularly relevant kind of CN is a Business Ecosystem (BE). A BE can be identified as a “long-term strategic alliance which tries to preserve local specificities, tradition, culture, and frequently benefit from local government incentives, involving a complex interplay of collaboration and competition around producers, consumers, regulators, and support entities” [7, 8].

One specific example of BE is given by the term Digital Business Ecosystem [3], which is also inspired on biological ecosystems, but with a stronger emphasis on the perspective of technological support. Furthermore, the term Collaborative Business Ecosystem (CBE) was introduced to emphasize the “collaborative environment” perspective [9–11]. The CBE is based on Collaborative Networks discipline and supports enterprises with collaborative tools to assist on the mitigation of organizations’ weaknesses while strengthen their know-how. As a result, organizations offering composite (integrated) care services in the ECE is expected to leverage their competitive advantage to customer agreement.

Our concept of Elderly Care Ecosystem (ECE) represents a particular case of a CBE. It includes various elements of a collaborative environment (administration, broker, virtual organization, planner, and coordinator), and specific elements that characterize it as an “Elderly Care Collaborative Network”, specifically the customers

(seniors), customer’s request and requirements, customer’s care needs, ECE services, and ECE service providers, including others [11–13]. Figure 1 presents the ECE environment domain diagram highlighting four ECE subsystems: ECE Manager System, ECE Information System, ECE Personalization System, and ECE Evolution System, and the three phases involved in the operationalization of ECE: Preparation phase, Execution phase, and Monitoring phase.

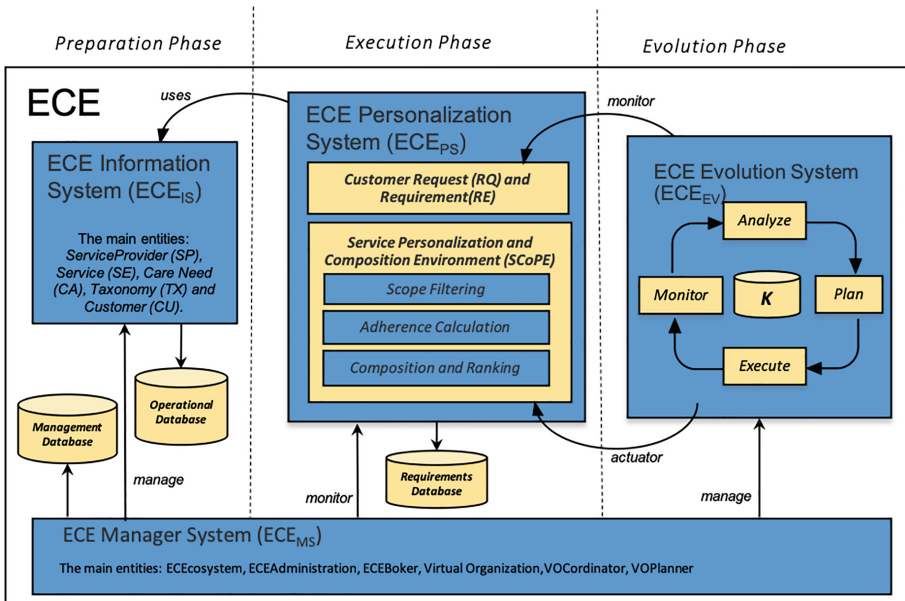


Fig. 1. ECE environment domain diagram

The *Preparation phase* corresponds to the creation of ECE and definition of its rules and functionalities within a collaborative environment. It involves (i) representing the main body of information and knowledge, (ii) the target audience identification, the considered stakeholders, specifically partners in the several areas (entity of support, entity of regulation, private institutions, governmental companies, freelancing professionals, caregivers, etc.) which are members of ECE, (iii) human and ICT resources, (iv) business and management rules; and (v) characterizing the available services. Based on a number of templates this phase involves creating the taxonomy of care need goals, identifying the service provider profile, service profile, and customer profile. The main subsystems responsible for the *Preparation phase* are the *ECE Manager System* (ECE_{MS}) and the *ECE Information System* (ECE_{IS}). ECE_{MS} is consolidated through the pillar of collaborative networks and the ECE_{IS} is more detailed in the next sections in the current chapter.

The *Execution phase* relates to the process of composition and personalization of services, including the ranking of the offered pairs (services and demand (customer care needs)). The main actuator subsystem at this stage is the *ECE Personalization System* (ECE_{PS}) which involves the Service Composition and Personalization Environment

(SCoPE method). This method comprises three steps: scope filtering, service adherence calculation, and service composition and ranking.

The Service Composition and Personalization Environment (SCoPE) was proposed in our previous work [5]. The method is composed of three main stages: (1) Scope Filtering – responsible for matching and filtering service and providers considering the care needs taxonomy, (2) Adherence Calculation – responsible for the outcomes of the first rating of services and providers considering their fitness for a specific customer, and (3) Service Composition and Ranking – responsible for recommending a solution list of providers. Figure 2 presents the SCoPE overview.

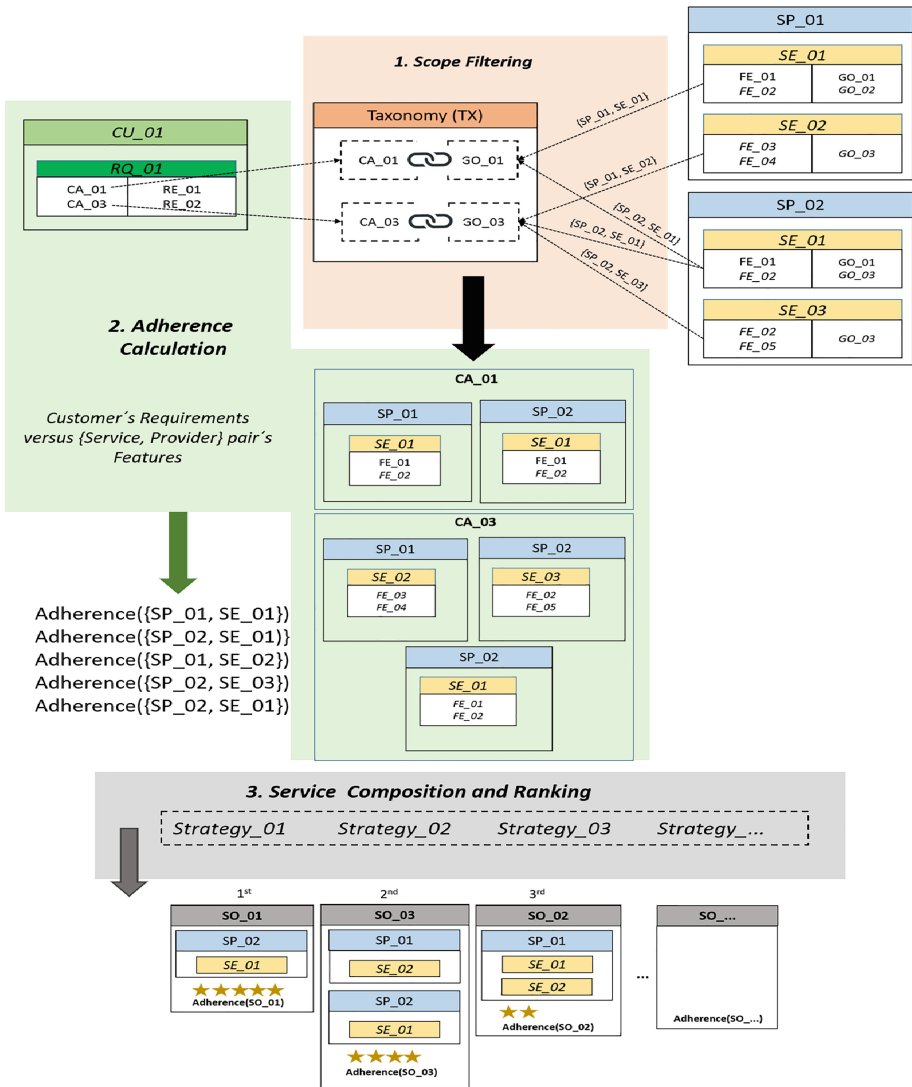


Fig. 2. SCoPE method overview

The *Monitoring phase* implements the *ECE Evolution System* (ECE_{EV}) that supports the service evolution and monitoring in the *ECE Service Personalization* (ECE_{PS}) environment. ECE_{EV} supports the Service Evolution (SEvol) method. Considering the dynamic environment and stages of life of the elderly, the *ECE broker* analyses the situation and adapts the services to fit each new context. In this way, SEvol evolves an existing care solution to cope with the new customer's life stage (e.g. handling new customer inputs, new or obsolete care needs, technological issues, new business strategies of service providers, etc.). The detailed process of the self-adaptive system approach for service evolution into ECE (ECE_{EV} and the SEvol method) is presented in Sect. 3.

3 ECE Evolution System

3.1 Evolutionary and Adaptive Systems

Adaptive and self-adaptive systems are a broad area of research with significant recent advances [14, 15]. These systems are characterized by having the capability of modifying their behavior and/or structure in response to their perception of the context and the system itself, and their requirements [16]. In summary, an adaptive system is “required to monitor itself and its context, detect significant changes, decide how to react, and act to execute such decisions” [14, 15].

The critical fact in a self-adaptive system is that its life-cycle should not be interrupted after its development and initial set up. Similarly, service operation should continue after deployment while evaluating the system and responding to changes at all time. In this sense, self-adaptive systems are realized as closed-loop systems with feedback loop [14]. During the adaptive process, it is possible to perform (if necessary) a human-in-the-loop action and the process continues after customer's feedback. In this situation, the adaptive system has a semi-dynamic adaptation.

Semi-dynamic adaptation is classified into two main paradigms that determine the range of possible states a system considers during the decision process [17, 18]: dynamic behavior adaptation, and dynamic reconfiguration.

In the case of dynamic behavior adaptation a system recognizes new environment conditions not envisioned during development and then control and order is emergent rather than predetermined. In the case of dynamic reconfiguration it encompasses possible variants of behavior that are somehow predefined before execution. During execution, current state, environment, and context are evaluated, and the most appropriate behavior variant is selected.

Some architecture-based adaptation frameworks have been proposed and developed over the years. They represent either academic or industry initiatives to address issues on the self-* properties and the adaptation process itself [15, 19]. Developing adaptive technologies and frameworks is beyond the scope of this work, hence existing adaptive approaches are considered to support our approach.

The proposed evolution system is based on the MAPE-K control loop structure [20, 21]. This control loop traditionally covers four elements: Monitor, Analyze, Plan and Execute. “Monitor” collects the details from the managed resources (e.g., sensors data,

customer's information, configuration property settings, etc.). The monitor function gathers information, filter it and aggregate it (besides normalizing) until detect a condition that requires analysis. "Analyze" function is responsible for complex data analysis and reasoning based on the conditions detected by monitoring phase. If some modification is necessary, a change request is invoked to the plan function. The "Plan" organizes the actions needed to achieve the target goals and creates (or selects) a process that will enact the modifications. Finally, "Execute" functions operationalize the modifications by executing the planned actions on managed resource. This is accomplished with the use of specific effectors.

3.2 SEvol Method

In the elderly care ecosystem domain, and for each new context change, the proposed ECE Evolution System analyses the situation (in collaboration with the relevant stakeholders) and adapts the service to fit that new context. In other words, the Service Evolution (SEvol) method supports the solution evolution to cope with the new life stage. Under this perspective, the notion of evolutionary service [22–26] means that the provided service is adapted to the senior's needs, and to any changes that affect the senior's life context.

Following MAPE-K, the SEvol method is based on a control loop composed of four main stages: (i) monitoring events that occur in the surrounding physical and social context (i.e., both context changes and messages exchanged between stakeholders); (ii) analyzing monitored data against solution requirements to identify need of adaptation; (iii) devising an evolution strategy that reconciles current solution with a new customer's context; and (iv) enacting such strategy while minimizing disturbances caused by suggested solutions. These stages are identified in the *i** rationale strategic model (see Fig. 3) that provides an intentional description of processes in terms of process elements and the rationales behind them [27].

The main actor is *Evolution System* (A-04 in Fig. 3) that is supported by additional elements of the model: *Context sensor* (A-01), *Agent* (A-02), and *Contextual actuator* (A-03). In more detail:

- *Context sensor* (A-01) is seen as a computational entity (hardware and software) providing raw data about the elderly environment. For instance, a bracelet that determines the current location of the customer or other stakeholders (e.g., who deliver/execute the care service), the sensor that determines the temperature and humidity levels in specific places, the smart communicator's automatic incoming/outgoing calls, etc.
- *Agent* (A-02) represents each of the actors who need to be monitored to ensure that they deliver according to their role in the ecosystem and send feedback about their acts. These agents may represent a senior, her/his guardian or caregiver, the coordinator of the virtual organization, VO (who manages the care service delivery for this senior), a service provider (which is part of a VO), etc. *Agents* are linked to the *Evolution System* (A-04) through inputs provision to identify a new request or through choices made in the human interaction. For instance, a substitution of a resource may be solicited by a service provider.

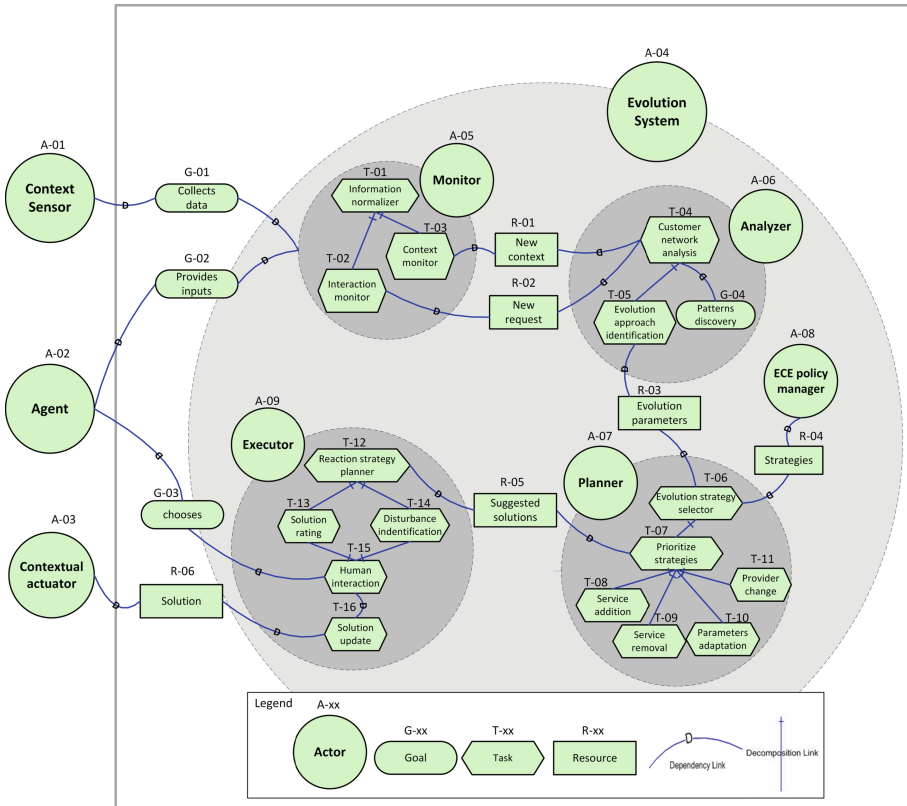


Fig. 3. Adapted i* rationale strategic model for the evolution system loop in ECE

– *Evolution System* (A-04) provides the self-adaptation capabilities of our model. This actor is split into four sub-actors:

Sub-actor *Monitor* (A-05) receives the information from agents (*Agent* (A-02)) and sensors (*Context Sensor* (A-01)). The inputs from the agents can be of several origins, for instance, from the customer and his/her family and guardian, or from the ECE, mainly originated in the Virtual Organization (VO) coordination, ECE management, or service provider. The inputs from sensors represent data about the elderly environment, for instance, information about senior’s sleep analysis. Examples of outcomes are (i) the identified new care need, (ii) indication that a care need is no longer present, and (iii) indication that service changes the delivery parameters.

Sub-actor *Analyzer* (A-06) receives, from *Monitor* (A-05), information about the current elderly context living (*New context* (R-01) or *New request* (R-02)) and observes the current pattern identifying the solution parameters that need to evolve.

Sub-actor *Planner* (A-07) selects evolution strategies to be adopted by the ECE policy manager (A-08), and ranks suggested solutions. The proposed solution evolution approach in ECE is based on (1) composition (or decomposition) of the current solution

(*Service addition* (T-08) or *Service removal* (T-09)); (2) solution parameters change (*Parameters adaptation* (T-10)), for instance, delivery conditions; or (3) the change of the entity responsible for the care service delivery (*Provider change* (T-11)).

Sub-actor *Executor* (A-09) changes the behavior of the managed resource using effectors based on the actions recommended by the *Planner* (A-07). Notice that evolution should not be considered a new personalization since it does not seek the better possible results from scratch, but instead seeks a satisfactory solution with the least possible disturbance to the customer (that is already used to the specific characteristics of current solution).

More details about these four sub-actors are presented in the following subsections.

Sub-actor Monitor. The *Monitor's* (A-05) purpose is to identify relevant changes in the physical and social context, notifying the *Analyzer* actor (A-06). To collect inputs, the *Monitor* relies on context sensors and agents. Distinct tasks are needed to carry out the *Monitor's* goal:

The task *Information normalizer* (T-01) initiates the monitoring function, taking its input from sensors or agents. The collected data and provided inputs are normalized to a “common language” that expresses the information on a context model (see Fig. 4).

The *Collects data* (G-01) and *Provides inputs* (G-02) are required by the *Monitor's* tasks: *Interaction monitor* (T-02) and *Context monitor* (T-03).

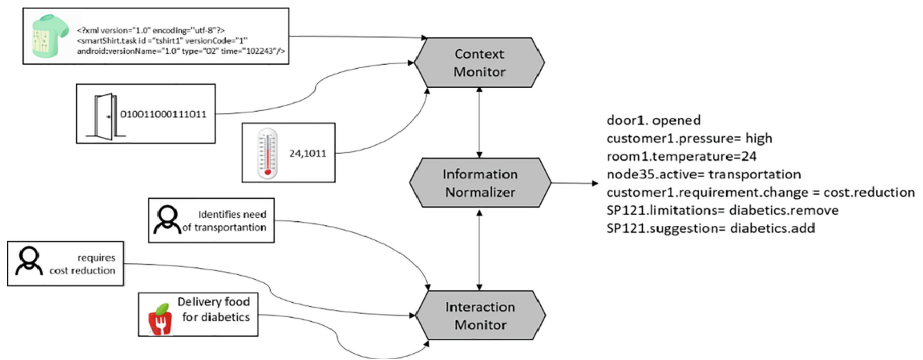


Fig. 4. Overview of *Information normalizer* task

The *Interaction monitor* (T-02) processes the status of existing non-standard data and exposes it through *New request* (R-02). For instance, a service provider will no longer deliver a specific service, or the customer wants to lower the price of a solution.

The *Context monitor* (T-03) computes information related to context and establishes *New context* (R-01). For example, if the customer has a medical appointment and her/his location is not moving, this information is sent (He/she did visit the doctor), or he/she is five days without leaving home, not participating in routine meetings (indicating signs of isolation, a possible new care need.)

Figure 4 sketches how the *Information normalizer* (T-01) works. Some inputs are sent in different formats: an XML file from the smart t-shirt, binary raw data from the door, CSV (comma-separated values) data from the thermometer, change of request from the customer activating (or disabling) care needs, customer is changing the constraints, and a service provider removing a limitation or adding an application suggestion.

Normalization demands the definition of a translation plan for each raw data format. If the task sources provide data in standard formats (e.g., XML file from the smart t-shirt), transformation schemes can be defined using a transformation language (e.g., XSLT – eXtensible Stylesheet Language Transformation).

If the house entrance door is closed (e.g. `door.status = closed`) and an event such as `open(door, time i)` happens, the *Context Monitor* (T-03) converts these data in terms of a shared context environment changing the status of the entrance door to `open (door.status = open)` – representing a *New context* (R-01): “`door1.opened`”. At the other hand, if the customer identifies a new need (e.g. need of transportation), the *Iteration monitor* (T-02) processes events related to context and exposes this requirement - representing a *New request* (R-02): “`node35.active = transportation`”.

Sub-actor Analyzer. *Sub-actor Analyzer* (A-06) is responsible for checking current information about the *Evolution System* (A-04) collected by the Monitor.

The role of requirements to the *Customer network analysis* (T-04) is to specify the goals that should (or should not) be achieved by certain agents, the plans these agents can execute, and the domain assumptions that must not be violated. In this sense, the more detailed requirements model the more accurate diagnosis will be. Moreover, agents’ granularity is constrained by pragmatic and technological issues. Identifying if the senior is lying on the bed for a long time is trivially gauged (e.g. with the use of pressure sensors), while effectively identifying if the senior is taking the medicine according to the current prescription is more complex.

The goal *Patterns discovery* (G-04) identifies and sets patterns of something routine that was not previously declared. It can declare patterns from the provider or stakeholders (sensors, caregivers, etc.). The task *Evolution approach identification* (T-05) observes the patterns of customer and indicates evolution parameters according to their relevance level, and provides the resource *Evolution parameters* (R-03) to sub-actor *Planner* (A-07).

Sub-actor Planner. *Planner* (A-07) analyses *Evolution* parameters (R-03) selecting the evolution strategies to this context serving as a system interface with ECE policy manager (A-07) that handles policies defined by ECE managers. *Evolution strategy selector* (T-06) receives the *Strategies* (R-04) from *ECE policy manager* (A-07) and prioritizes these solutions considering the ECE policies and goals (subtask *Prioritize strategies* (T-07)). The suggested strategies for solution evolution are based on the following tasks: *Service addition* (T-08), *Service removal* (T-09), *Parameters adaptation* (T-10), and *Provider Change* (T-11). In the end, a resource *Suggested solutions* (R-05) is sent to the sub-actor *Executor* (A-09).

In our work, the evolution (or adaptation) is mainly based on direct inputs of customer. We consider as primary inputs to the evolution process: (a) the identified new care need, (b) identification that a care need is no longer present, or (c) identification that service changes the delivery conditions.

The proposed service evolution strategy in ECE is based on composition (or decomposition in case of service removal) of the current solution or the parameter change of delivery conditions. For each primary input, the detailed strategy is presented below (more details in [25]).

- **Situation (a): adding a care need x .** The newly added care need is covered by current solution of the customer; therefore, adding a new care need implies the adaptation of the integrated {service, provider} pairs. It is possible to classify this adaptation into two categories: (a1) Identifying (in the current solution) a {service, provider} pair that covers the new care need (the solution is not changed). (a2) Adding a new {service, provider} pair that covers the new care need. So the process should identify if the current solution satisfies the new care need. If so, the process ends. Otherwise, the service and provider fragments which cover the new care need x are identified in order to extend the current solution (based on the adherence level resulting from SCoPE method).
- **Situation (b): removal of a care need x .** This removal should not affect the current solution for the other care needs. So, two cases are considered: (b1) Removal of {service, provider} pair that covers the x care need (if this pair does not cover any other need). (b2) Change of {service, provider} pair that jointly covers x and other care needs (for instance a care need y). The immediate removal of a {service, provider} pair fragment (without prejudice to current solution) can only be done if it is exclusively attending the x care need. In this situation (b1), the {service, provider} pair fragment is eliminated along with the obsolete care need, and the calculation of the solution adherence is updated. Otherwise (b2), the {service, provider} pair fragment that is attending care needs x and y goes through a new process of calculating the adherence (SCoPE method) considering now only the care need that stays (y). The fragment can be updated if there is a better service adherence to y . The adherence is calculated by the SCoPE process. This process can be repeated when other care needs are also covered by the same fragment.
- **Situation (c): modifying parameters of a care need.** In this situation, the customer's care needs remain the same. However, specific requests are modified in ECE. For example, the customer usually requires a transportation service once a week, but for the next month, it will be twice a week (frequency parameter); the customer had a collective transportation service, but now she/he wants private transportation (service features parameter), etc. The evolution plan to change a care need parameter involves two stages: (c1) Identify the parameter which should be changed checking if the new value is available for the current solution, (c2) Find a {service, provider} pair available that attends the new parameter.

Sub-actor Executor. The sub-actor Executor (A-09) executes the action and the solution is adapted according of new requirements. The task Reaction strategy planner (T-12) calculates the rating of the evolutionary solution (task Solution rating (T-13))

and identifies the disturbance in relation to the old solution (task Disturbance identification(T-14)). In the next step, the human-in-the-loop phase is started and the ECE broker and the customer confirm (or decline) the new proposed solution (task Human interaction (T-15)), and the solution is updated (task Solution update (T-16)).

4 ECE Processes

As previously mentioned in Sect. 2, the ECE environment supports three main processes associated to care services: Preparation, Execution and Monitoring, and uses four environments: ECE Information System and ECE Manager System (briefly presented Sect. 2), ECE Personalization System and ECE Evolution System (presented in the Sect. 3).

To demonstrate the integration of environments and their roles step by step a workflow diagram is presented in Fig. 5. This diagram illustrates the main process flow split in five lanes (ECE Manager System, Customer, Service Provider, ECE Personalization System and ECE Evolution System). A brief description of each process of Fig. 5 is presented in the Table 1.

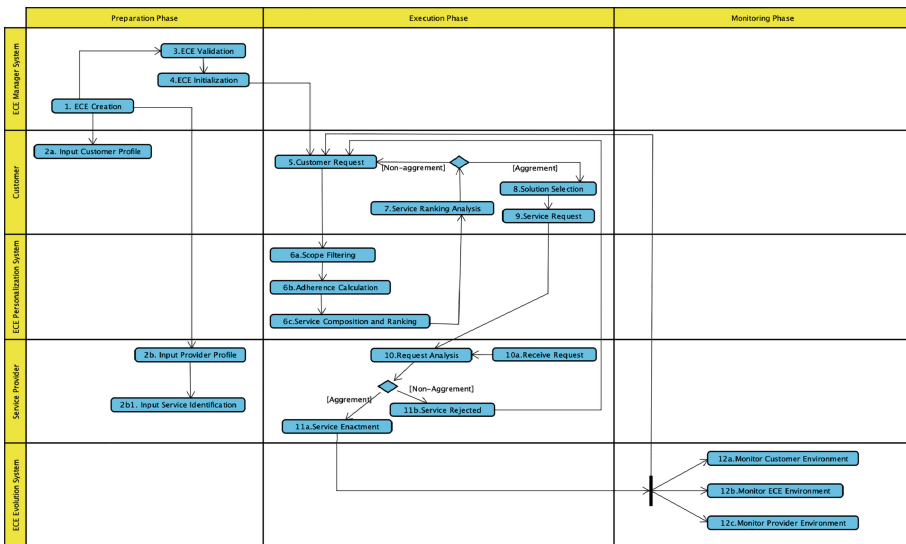


Fig. 5. ECE workflow diagram divided into five lanes

To illustrate the applicability of the proposed ECE framework, Table 2 presents a summary of a practical application scenario relating the main ECE process.

Table 1. Brief description of each sub-process of Fig. 5

Phase	ID	Activity	Lane	Description
Preparation	1.	ECE Creation	ECE Manager System	ECE definition of its rules and functionalities. Characterization of ECE elements profile templates identifying the proposed care needs taxonomy.
Preparation	2a.	Input Customer Profile	Customer	Customer profile implementation. Registering a customer.
Preparation	2b.	Input Provider Profile	Provider	Service provider profile implementation. Registering a service provider.
Preparation	2b1.	Input Service Identification	Provider	Service profile implementation. Registering a service linking with the provider.
Preparation	3.	ECE Validation	ECE Manager System	Validation of all the elements to start the execution process. There is at least one item in each profile (or a minimum established in the ECE Creation).
Preparation	4.	ECE Initialization	ECE Manager System	Release to wait a customer's input.
Execution	5	Customer Request	Customer	Filling out customer choices and preferences. Collecting of data for the service request.
Execution	6a.	Scope Filtering	ECE Personalization System	Filtering of potential services to attend customer request based on care need taxonomy.
Execution	6b.	Adherence Calculation	ECE Personalization System	Calculating the adherence between customer request and potential services found in scope filtering activity.
Execution	6c.	Service Composition and Ranking	ECE Personalization System	Building the matrix of proposed solutions to customer ranking by service adherence.
Execution	7.	Service Ranking Analysis	Customer	Analyzing of the proposals presented if it is in accordance with what the customer intends (in the-human-in-the-loop action). If not, return to the initial form of execution form (activity 5. Customer Request). If so, proceed to Service Selection.
Execution	8.	Solution Selection	Customer	Human-in-the-loop for confirmation of selected solution.
Execution	9.	Service Request	Customer	After customer favourable evaluation, approval is given to submit the solution to the involved service providers.
Execution	10.	Request Analysis	Provider	The business partners (which are member of the CN) respond to the customer request.
Execution	10a.	Receive Request	Provider	Provider analyses data and adjusts business strategy if necessary.
Execution	11a.	Service Enactment	Provider	Delivery of Service Solution.
Execution	11b.	Service Rejected	Provider	There was no agreement between the partners in the provision of the service. Return to form 7. Service Ranking Analysis.
Monitoring	12a.	Monitor Customer Environment	ECE Evolution System	Identifying new care needs or obsolete care needs of the customer.
Monitoring	12b.	Monitor ECE Environment	ECE Evolution System	Promoting addition or removal of services, addition or removal of service providers, and ECE Manager is changing rules or strategies.
Monitoring	12c.	Monitor Provider Environment	ECE Evolution System	Service provider is doing update on its services.

Table 2. ECE process in a practical application scenario

ECE's Activities (main)	Scenario																																										
2a. Input Customer Profile	[[idCustomer:c1; name:Robert; gender:male; geographicalArea: Lisbon; guardian:[Marco:son]; limitations:[{diabetic, memory less, cardiac}; maximumInvestment:100; technologicalKnowledge:low]]																																										
2b. Input Provider Profile	[[idProvider:p1; name:ALLCorporation; geographicalArea:world; services:{s1, s2, s3}; rating:5;]; [idProvider:p2; name:EldCareCorporation; geographicalArea:Europe; services:{s1, s4, s5}; rating:4;]]																																										
2b1. Input Service Identification	[[provider:{idProvider:p1, idProvider:p2}), (service:{idService:s1, name: Security_Guard_service; careNeed: {diseaseMaintenance; high; safety: high}; p1.cost:40; p2.cost:30; area:independentLiving; ...})]; [[provider:{idProvider:p1}), (service:{idService:s2, name:MyMonitor careNeed: {safety; veryHigh}; cost:40; area:health; ...}); ({idService:s3, name:Agenda; careNeed: {diseaseMaintenance; low; safety: medium }; cost:40; area: independentLiving; ...}) [[provider:{idProvider:p2}), (service:{idService:s4, name:TotalControl; careNeed: {diseaseMaintenance; high;}; cost:100; area: health;}) ({idService:s5, name:HelpYou; careNeed: {diseaseMaintenance; medium; safety: high}; cost:75; area: health/independentLiving;})																																										
5. Customer Request	[[idCustomer:c1; careNeed:({diseaseMaintenance,high},{safety, veryHigh}); requirement:({cost.soft≤100};{tecnhologicalUsability.soft=,high})]																																										
6b. Adherence Calculation (more details in [5])	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>Service, provider</th> <th>diseaseMaintenance</th> <th>safety</th> <th><i>adherence</i></th> <th>Cost</th> <th>Cost-Benefit Ratio (%)</th> </tr> </thead> <tbody> <tr> <td>sp₁₁</td> <td>0.885</td> <td>0.622</td> <td>0.75</td> <td>40 €</td> <td>1.71</td> </tr> <tr> <td>sp₁₂</td> <td>0.184</td> <td>0.782</td> <td>0.48</td> <td>30 €</td> <td>1.60</td> </tr> <tr> <td>sp₂₁</td> <td>0.000</td> <td>0.927</td> <td>0.46</td> <td>40 €</td> <td>1.15</td> </tr> <tr> <td>sp₃₁</td> <td>0.345</td> <td>0.451</td> <td>0.39</td> <td>85 €</td> <td>0.46</td> </tr> <tr> <td>sp₄₂</td> <td>0.767</td> <td>0.000</td> <td>0.38</td> <td>100 €</td> <td>0.38</td> </tr> <tr> <td>sp₅₂</td> <td>0.639</td> <td>0.791</td> <td>0.71</td> <td>75 €</td> <td>0.95</td> </tr> </tbody> </table>	Service, provider	diseaseMaintenance	safety	<i>adherence</i>	Cost	Cost-Benefit Ratio (%)	sp₁₁	0.885	0.622	0.75	40 €	1.71	sp₁₂	0.184	0.782	0.48	30 €	1.60	sp₂₁	0.000	0.927	0.46	40 €	1.15	sp₃₁	0.345	0.451	0.39	85 €	0.46	sp₄₂	0.767	0.000	0.38	100 €	0.38	sp₅₂	0.639	0.791	0.71	75 €	0.95
Service, provider	diseaseMaintenance	safety	<i>adherence</i>	Cost	Cost-Benefit Ratio (%)																																						
sp₁₁	0.885	0.622	0.75	40 €	1.71																																						
sp₁₂	0.184	0.782	0.48	30 €	1.60																																						
sp₂₁	0.000	0.927	0.46	40 €	1.15																																						
sp₃₁	0.345	0.451	0.39	85 €	0.46																																						
sp₄₂	0.767	0.000	0.38	100 €	0.38																																						
sp₅₂	0.639	0.791	0.71	75 €	0.95																																						
6c. Service Composition and Ranking (more details in [5])	<table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>Solutions</th> <th>diseaseMaintenance</th> <th><i>Safety</i></th> <th><i>adherence</i></th> <th>Cost</th> <th>Cost-Benefit Ratio (%)</th> </tr> </thead> <tbody> <tr> <td><i>Solution₁</i></td> <td>sp₁₁</td> <td>sp₂₁</td> <td>0.906</td> <td>80 €</td> <td>1.113</td> </tr> <tr> <td><i>Solution₂</i></td> <td colspan="2">sp₅₂</td> <td>0.715</td> <td>75 €</td> <td>0.953</td> </tr> <tr> <td><i>Solution₃</i></td> <td>sp₁₁</td> <td>sp₅₂</td> <td>0.838</td> <td>115 €</td> <td>0.729</td> </tr> <tr> <td><i>Solution₄</i></td> <td>sp₄₂</td> <td>sp₂₁</td> <td>0.847</td> <td>140 €</td> <td>0.605</td> </tr> </tbody> </table>	Solutions	diseaseMaintenance	<i>Safety</i>	<i>adherence</i>	Cost	Cost-Benefit Ratio (%)	<i>Solution₁</i>	sp ₁₁	sp ₂₁	0.906	80 €	1.113	<i>Solution₂</i>	sp ₅₂		0.715	75 €	0.953	<i>Solution₃</i>	sp ₁₁	sp ₅₂	0.838	115 €	0.729	<i>Solution₄</i>	sp ₄₂	sp ₂₁	0.847	140 €	0.605												
Solutions	diseaseMaintenance	<i>Safety</i>	<i>adherence</i>	Cost	Cost-Benefit Ratio (%)																																						
<i>Solution₁</i>	sp ₁₁	sp ₂₁	0.906	80 €	1.113																																						
<i>Solution₂</i>	sp ₅₂		0.715	75 €	0.953																																						
<i>Solution₃</i>	sp ₁₁	sp ₅₂	0.838	115 €	0.729																																						
<i>Solution₄</i>	sp ₄₂	sp ₂₁	0.847	140 €	0.605																																						
8. Solution Selection	[[idCustomer:c1;] (Strategy:humanInTheLoop); (solution:Solution1: {sp11,sp22}, ad:0.906;cost:80)]																																										
12a. Monitor Customer Environment	newRequest:([idCustomer:c1; careNeed:({diseaseMaintenance,high}); requirement:({cost.soft≤50};{tecnhologicalUsability.soft=,high})]																																										
8. Solution Selection	[[idCustomer:c1;] (Strategy:serviceRemoval); (Solution:newSolution: {sp11}, ad: 0.885;cost:40)]																																										

5 Conclusions

In this paper, we present a view about the influence of evolutionary and adaptive systems identifying the MAPE-k control loop structure. We propose a service evolution method (SEvol) to support the adaptation process (evolution) of current customer's solution to new requests.

Following MAPE-K, the SEvol method is based on a control loop composed of four main stages: (Monitor) monitoring events that occur in the surrounding physical and social context; (Analyzer) analyzing monitored data against solution requirements to identify need of adaptation; (Planner) devising an evolution strategy that reconciles current solution with a new customer's context; and (Executor) enacting such strategy while minimizing disturbances caused by suggested solutions. The K corresponds to the current customer's solution. Providing an intentional description of processes in terms of process elements and the rationales behind them, the Evolution System was explained through an i* rationale strategic model. In the end, a workflow diagram is presented considering the main processes of ECE, demonstrating the roles of the ECE environments and the main stakeholders, separated in the three stages of: preparation, execution and monitoring. In addition, a practical application scenario is presented in the main ECE process to evaluate our approach. This proposal is intended to provide an adaptive system that can work well with an ECE framework to service personalization and evolution.

Ongoing developments include the assessment of the proposed method within a real elderly care ecosystem.

Acknowledgements. The authors acknowledge the contributions of the Portuguese FCT-Strategic program UID/EEA/00066/2019 for providing partial financial support for this work.

References

1. Kearney, A.T.: Understanding the Needs and Consequences of the Ageing Consumer (2013). The Consumer Goods Forum. <https://www.atkearney.com/documents/10192/682603/Understanding+the+Needs+and+Consequences+of+the+Aging+Consumer.pdf/6c25ffa3-0999-4b5c-8ff1-afdca0744fdc>. Accessed 10 Feb 2019
2. Fengler, W.: The End of the Population Pyramid (2014). <http://www.economist.com/blogs/graphicdetail/2014/11/daily-chart-10>. Accessed 10 Feb 2019
3. Gartner, I.: Gartner's 2018 Hype Cycle for Emerging Technologies Identifies Three Key Trends That Organizations Must Track to Gain Competitive Advantage (2018). <http://www.gartner.com/doc/2847417?refval=&pcp=mpe#a-1321928256>. Accessed 12 Jan 2019
4. Bureau, P.R.: 2018 World Population Data (2018). <http://www.worldpopdata.org/index.php/map>. Accessed 12 Jan 2019
5. Baldissera, T.A., Camarinha-Matos, L.M.: SCoPE: service composition and personalization environment. *Appl. Sci.* **8**(11), 2297 (2018)
6. Camarinha-Matos, L.M., Afsarmanesh, H.: Classes of collaborative networks. In: *Encyclopedia of Networked and Virtual Organizations*, I.S. Reference, Editor, USA, pp. 193–198 (2008)

7. Camarinha-Matos, L.M., Afsarmanesh, H., Boucher, X.: The role of collaborative networks in sustainability. In: Camarinha-Matos, L.M., Boucher, X., Afsarmanesh, H. (eds.) PRO-VE 2010. IAICT, vol. 336, pp. 1–16. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15961-9_1
8. Camarinha-Matos, L.M., Rosas, J., Oliveira, A.I., Ferrada, F.: A collaborative services ecosystem for ambient assisted living. In: Camarinha-Matos, L.M., Xu, L., Afsarmanesh, H. (eds.) PRO-VE 2012. IAICT, vol. 380, pp. 117–127. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32775-9_12
9. Camarinha-Matos, L.M.: Collaborative Business Ecosystems and Virtual Enterprises: IFIP TC5/WG5.5 Third Working Conference on Infrastructures for Virtual Enterprises ... in Information and Communication Technology). Springer, Heidelberg (2013)
10. Graça, P., Camarinha-Matos, L.M.: The need of performance indicators for collaborative business ecosystems. In: Camarinha-Matos, L.M., Baldissera, T.A., Di Orio, G., Marques, F. (eds.) DoCEIS 2015. IAICT, vol. 450, pp. 22–30. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16766-4_3
11. Baldissera, T.A., Camarinha-Matos, L.M.: Towards a collaborative business ecosystem for elderly care. In: Camarinha-Matos, L.M., Falcão, A.J., Vafaei, N., Najdi, S. (eds.) DoCEIS 2016. IAICT, vol. 470, pp. 24–34. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-31165-4_3
12. Baldissera, T.A., Camarinha-Matos, L.M.: Services personalization approach for a collaborative care ecosystem. In: Afsarmanesh, H., Camarinha-Matos, L.M., Lucas Soares, A. (eds.) PRO-VE 2016. IAICT, vol. 480, pp. 443–456. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45390-3_38
13. Baldissera, T.A., Camarinha Matos, L.M., DeFaveri, C.: Designing elderly care ecosystem in collaborative networks environment. In: International Conference on Computing, Networking and Informatics. IIEEE, Ota (2017). Editor
14. Salehie, M., Tahvildari, L.: Self-adaptive software: landscape and research challenges. *ACM Trans. Auton. Adapt. Syst.* **4**(2), 1–42 (2009)
15. de Lemos, R., et al.: Software engineering for self-adaptive systems: a second research roadmap. In: de Lemos, R., Giese, H., Müller, H.A., Shaw, M. (eds.) Software Engineering for Self-Adaptive Systems II. LNCS, vol. 7475, pp. 1–32. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-35813-5_1
16. Laddaga, R., Robertson, P.: Self adaptive software: a position paper. In: SELF-STAR: International Workshop on Self-* Properties in Complex Information Systems (2004)
17. Shelton, C.P., Koopman, P., Nace, W.: A framework for scalable analysis and design of system-wide graceful degradation in distributed embedded systems. In: Proceedings of the Eighth International Workshop on Object-Oriented Real-Time Dependable Systems, (WORDS 2003) (2003)
18. Macedo, P.A.P.: Models and Tools for Value Systems Analysis in Collaborative Environments. Universidade Nova de Lisboa, Monte da Caparica (2011)
19. Brun, Y., et al.: Engineering self-adaptive systems through feedback loops. In: Cheng, B.H. C., de Lemos, R., Giese, H., Inverardi, P., Magee, J. (eds.) Software Engineering for Self-Adaptive Systems. LNCS, vol. 5525, pp. 48–70. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02161-9_3
20. IBM: An architectural blueprint for autonomic computing. In: IBM White Paper (2006)
21. Arcaini, P., Riccobene, E., Scandurra, P.: Modeling and analyzing MAPE-K feedback loops for self-adaptation. In: Proceedings of the 10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems. IEEE Press (2015)
22. O’Grady, M.J., Muldoon, C., Dragone, M., Tynan, R., O’Hare, G.M.: Towards evolutionary ambient assisted living systems. *J. Ambient Intell. Hum. Comput.* **1**(1), 15–29 (2010)

23. Hong, J., Suh, E.-H., Kim, J., Kim, S.: Context-aware system for proactive personalized service based on context history. *Expert Syst. Appl.* **36**(4), 7448–7457 (2009)
24. Brown, A., Johnston, S., Kelly, K.: *Using Service-Oriented Architecture and Component-Based Development to Build Web Service Applications*. Rational Software Corporation, San Jose (2002)
25. Baldissera, T.A., Camarinha-Matos, L.M.: Services evolution in elderly care ecosystems. In: Camarinha-Matos, L.M., Afsarmanesh, H., Rezgui, Y. (eds.) *PRO-VE 2018. IAICT*, vol. 534, pp. 417–429. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99127-6_36
26. Marcos-Pablos, S., García-Peñalvo, F.J.J.S.: Technological ecosystems in care and assistance: a systematic literature review. *Sensors* **19**(3), 708 (2019)
27. Yu, E.S., Mylopoulos, J.: *From ER To “aR”—modelling strategic actor relationships for business process reengineering*, pp. 125–144. Ph.D. thesis. University of Toronto, Toronto – Canadá (1995)