

Dynamic Homecare Service Provisioning: A Field Test and Its Results

Alireza Zarghami, Mohammad Zarifi, Marten van Sinderen, and Roel Wieringa

Department of Electrical Engineering, Mathematics and Computer Science,
University of Twente, Enschede, The Netherlands
{a.zarghami,m.zarifi,m.j.vansinderen,r.j.wieringa}@utwente.nl

Abstract. Providing IT-based care support for elderly at home is proposed as a highly promising approach to address the aging population problem. With the emergence of homecare application service providers, a homecare system can be seen as a linked set of services. Configuring and composing existing homecare application services to create new homecare composite applications can reduce the application development cost. The idea even looks more promising if the service provisioning is dynamic, i.e., if applications can update their behaviors with respect to the contextual changes without or with minimum manpower. Dynamic service provisioning can play an important role to accept homecare systems in practical settings. This motivated us to develop a Dynamic Homecare Service Provisioning (DHSP) platform to address the homecare context changes in an effective and efficient manner. As a proof of concept, we have developed a software prototype of our platform. The prototype was subsequently used in a real-world field test at a care institution in the Netherlands to validate the approach. This paper describes the design of the field test and reflects on the outcome of the validation experiments.

1 Introduction

The population of many Western-European countries have an increasing percentage of elderly people and, consequently, the healthcare-related cost for these countries is growing [4,10]. Providing automated care support for elderly people in their own home is proposed as a highly promising solution to alleviate the problem of the growing healthcare related cost for elderly [7]. A homecare system is "a *potentially linked set of services ... that provide or support the provision of care in the home*" [12]. With the emergence of homecare application service providers, which provide IT-based healthcare services (e.g., blood pressure measurement service) for elderly people at home, a homecare system can be seen as a linked set of IT-based services, i.e., application services.

The time and cost of homecare application development and provisioning play an important role to have widely-accepted practical homecare systems. With the emergence of the Service-Oriented Architecture (SOA) paradigm, composing applications by reusing existing application services has become technically and economically feasible. The used application services can be provided by different healthcare-related organizations, and programmers can use them to build

a composite homecare application without knowing their internal implementation mechanisms. The idea even looks more encouraging if service provisioning can be dynamic, if these composite applications can update their behaviors with respect to the contextual changes without or with minimum manpower.

People needing homecare applications, such as patients or some elderly people, are generally called care-receivers. They can use their systems in their own ways. Thus many (un)foreseen changes may occur during the provisioning of a homecare application. Care-receivers are subject to different types of contextual changes such as changes in their vital-signs (e.g., blood pressure value), location (e.g., inside or outside home) and their capabilities (e.g., sight, hearing). These changes motivate the use of dynamic service provisioning in the homecare domain.

Dynamic homecare service provisioning is not easy to realize because (1) it must be cheaper (both human and system resources) than the current solution (otherwise the homecare domain stakeholders are not interested), (2) each user must be independently and adequately supported (otherwise the end-users (e.g., care-receivers) are not interested), (3) safety of the care-receivers may not be at risk (otherwise the government and the general public will be opposed). For instance, if the application must send an alert to a care-giver such as nurse, failure in sending that alert could be life-threatening in some specific circumstances.

Our top-level goal is to design a Dynamic Homecare Service Provisioning (DHSP) platform to address the homecare contextual changes in an effective and efficient manner. Although several promising prototypes have been done in academia [11,13,1,5], yet there are many challenges and hurdles to overcome before having a realistic homecare solution [12,9]. Based on our study, we observed that existing solutions have not addressed dynamic provisioning and its requirements as much as other challenges such as distributed and heterogeneous application service providers [23]. Specially, with respect to homecare requirements such as compliance with medical protocols, safety-criticalness and limited human resources (lack of care-givers), a homecare-specific dynamic service provisioning (we call it DHSP) platform is required [22].

We propose a DHSP platform which has been prototyped and tested with real care-givers, care-receivers and homecare application service providers. The contribution of this paper is threefold: a) we describe the design and execution of a field test, b) we present the collected and analyzed data from the field test, and c) we report on interesting results obtained from the field test and considered useful to improve similar systems. The field test is designed in two experiments and the goal of doing them is to study the usability of the platform in terms of effectiveness, efficiency and the end-user's satisfaction.

The rest of the paper is structured as follows. In Section 2, we introduce our DHSP and discuss related concepts. In Section 3, we describe an evaluation strategy for evaluating our approach. Section 4 presents the setup of the field test and the role of involved organizations (including service providers and the care center). Section 5 presents the results of the first experiment, and how the system evolved based on the changes requested by the care-givers. In Section 6

we discuss the results of the second experiment, how the system improved, and the lessons learned from the field test. In Section 7 we conclude the paper with an outlook on our future research.

2 Dynamic Homecare Service Provisioning Platform

We define a DHSP platform as an *adaptive*, *tailorable* and *evolvable* application service provisioning platform. In dynamic service provisioning, a composite application can be reconfigured. This can happen automatically on-the-fly (called *adaptive* service composition), by end-users such as nurses (which we call *tailorable* service composition) or by a programmer (which we call *evolvable* service composition).

Several dynamic service provisioning approaches have been proposed to facilitate the adaptation of composite applications with respect to contextual changes [14,16]. These changes can be *foreseen* or *unforeseen*. Automated reconfiguration of services must be foreseen at design time, and the composite application must be able to monitor relevant changes and adapt to them at run-time based on predefined application logic (adaptivity). Tailored reconfiguration is different: It is the end-user who recognizes the condition for change, but the possible responses to this change have been predefined and can be configured by the end-user (tailorability). In contrast, unforeseen changes are not known at design time and no possible responses have been planned at design time. Thus, at a later phase in the lifecycle, a programmer of a composite application must modify the application logic to deal with unforeseen changes (evolvability) [3]. The aim of our research is to provide the infrastructure for all three forms of dynamicity in a composite service-oriented application in the homecare domain [20].

Fig. 1 shows how an application is defined and executed on the DHSP platform. To decouple the concerns, we assume that there is a separate tailoring platform that takes care of creation and tailoring of service plans to be executed by the provisioning platform. A homecare system as we propose consists of a tailoring platform, a provisioning platform and several application services. A *service plan* consists of one or more *service building blocks* (SBBs) and describes the configuration and orchestration of instances of these service building blocks with respect to run-time circumstances. The service plan specifies the behavior of a composite application at runtime by specifying which application service will be selected for a SBB and how the selected service will be configured and orchestrated [21]. Each service plan can have several predefined orchestrations (of SBBs) of which one is selected at runtime based on decision making rules. A SBB defines a set of functionalities at the abstract level that can be implemented by alternative application services. The SBBs have several configurations that determine which application service should be selected. For instance, a SBB of medicine dispensing can be implemented by different medicine dispenser devices which might be provided by different vendors.

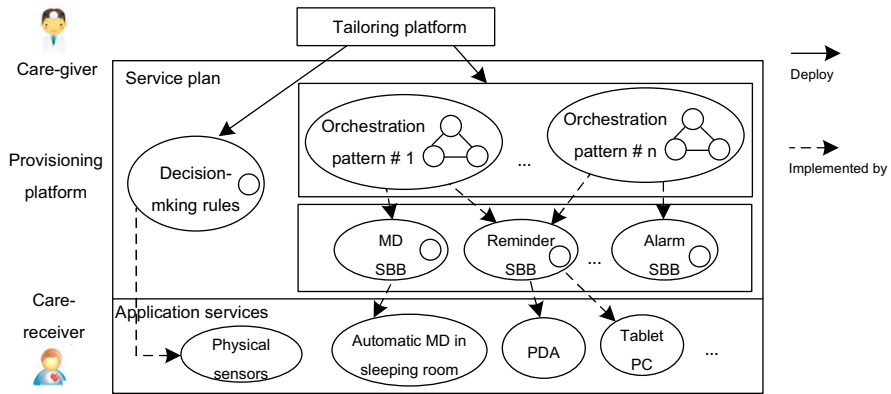


Fig. 1. How a service plan is defined and executed on our DHSP platform

To improve the evolvability, we separate these decision making rules from the orchestrations as a decision service [20]. The decision service can be called by the orchestration patterns. The decision service employs physical sensors through their corresponding application services. Based on the data coming from the physical sensors, the output of decision service determines which orchestration and configurations must be selected.

3 Validation Criteria

We want to improve homecare systems by facilitating dynamic service provisioning. The improvement criteria can be classified as: (a) To adapt a homecare application successfully within a certain time expected by an end-user, (b) To deploy a (re)taiored homecare application successfully within a certain time expected by a care-giver, and (c) To deploy an evolved homecare application successfully within a certain time expected by a programmer.

To investigate the needed time, we need to measure the efficiency of the platform and the efficiency correlates with the level of achieved effectiveness [8]. In the literature, usability is defined as the software quality attribute which comprises the efficiency, effectiveness and satisfaction aspects [8]. Therefore, We have performed a field test to evaluate the usability of the DHSP platform. Usability is a multidimensional characteristic which must be measured in the context of users performing a task with a system in a specific environment [2]. Most usability evaluation methods gather both subjective and objective quantitative data. Subjective data are measures of participants' opinions or attitudes concerning their perception of usability. Objective data are measures of participants performance, such as deployment completion time and successful rate [15].

To evaluate the usability of our DHSP platform, we follow the ISO 9241-11 definition of usability, which is "Extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction

in a specified context of use” [8]. The standard provides guidelines to measure the effectiveness and efficiency, which results in objective data (through system transaction logs), and to measure satisfaction, which delivers subjective data (through end-users’ interviews). Being able to combine objective and subjective measurements, would be useful to know which level of effectiveness and efficiency is acceptable in the homecare domain. Moreover, we identified explanations of our observations that allowed us to understand which parts of our approach need further improvement.

3.1 Effectiveness

The ISO standard defines effectiveness as *”accuracy and completeness with which users achieve specified goals”*. In our case, we define it as the number of system tasks such as sending an alert (adaptivity goals), deploying a service plan (tailorability goals) or modifying the application logic (evolvability goals), which have been completed within a specific time without systems errors. Since homecare systems are real-time reactive systems [17], task completion is measured time-dependently. For instance, sending a late alert or sending a reminder when it is not needed, is considered an error, although the system task is completed. Moreover, we exclude the care-givers’ mistakes. For instance, if a care-giver makes a mistake in tailoring an application but the deployment process goes well, we consider it as a completed and successful task. The effectiveness can be scored on a scale of 0 to 100%. In our field test, to measure the effectiveness, we will measure successful system task completion rate, i.e, the number of the system tasks that have been completed within a specific time as a percentage of the total number of system tasks.

$$\begin{aligned} \text{successful - system - task - completion - rate} &= \\ &= \frac{\text{completed-system-tasks} * 100}{\text{total-number-of-system-tasks}} \end{aligned}$$

3.2 Efficiency

The ISO standard defines efficiency as *”the level of effectiveness achieved to the expenditure of resources”*. In our case, we define it as the completion time of the system tasks. To evaluate the completion time, we should compare it with other homecare systems. Since in our case there are no other systems to compare, we evaluate the completion times with the care-givers’ expectations to see if the completion times are acceptable or not.

3.3 Satisfaction

The ISO standard defines satisfaction as *”the extent to which users are free from discomfort, and their attitudes towards the use of the product”*. In our case, we define it as the perceived effectiveness and perceived efficiency of the system from the end-users’ point of view. The satisfaction has some other aspects which

are more related to interaction with the end-users [6], however, our platform interacts indirectly with the end-users through the application services or the tailoring platform. Thus, to evaluate the satisfaction of the DHSP platform, we only consider the perceived effectiveness and efficiency. Moreover, we are interested to see how satisfaction can be affected by using different application services.

In order to measure satisfaction, we used questionnaires. Since we limit the satisfaction to the perceived effectiveness and efficiency, the existing usability questionnaires, that takes some other aspects into account [6], do not suit our requirements. Therefore, we have designed a questionnaire to ask specific questions about the perceived effectiveness and efficiency for each type of system task. Our questionnaire contains 8 open-end attitudinal questions to uncover end-users' beliefs and thoughts on the effectiveness and efficiency of each type of system task.

These questions are task-specific and designed to measure effectiveness and efficiency of each system task. For instance, (1) does the system remind the care-receivers on time to measure their vital signs or to take their medications?, (2) After measuring the vital signs, does the system show the measured values on the Tablet PC in a reasonable time?, and (3) Does the system stop sending reminders after the care-receiver has measured his/her vital signs? Each question has two answer parts: (1) *answer (yes/no)* and (2) *Comment on the question*. Due to the limited space, we can not explain all the questions in this paper.

4 Setup of the Experiments

The field test has been done in two experiments of two months each, with one month in between to improve existing applications and to add new applications. The field test is an action case study [19], in which we aim to improve the current situation of providing care by using a DHSP platform. We follow the guidelines described by Wieringa in [18] to perform the experiments systematically.

Each series of the experiments was conducted in a near real-world setting in a care institute in the Netherlands and each series lasted for two months. The experiments are close to a real-world setting, because some real-world aspects are present, such as real care-receivers, real institution, real nurses and realistic scenarios, but other aspects are absent, such as only a single homecare institution, a few users and candies instead of real medicines. In this section, first, we explain the scenarios to be used in the experiments, then we describe which actors participate in the experiment and the role of each one (e.g., which service provider provides which application service) and finally, we explain the measurement tools and how we collect data.

4.1 Homecare Applications

In our field test, we have three types of homecare applications: 1) vital-sign monitoring (VsM), 2) medication monitoring (MdM) and 3) social activity

monitoring (SaM). For VsM, we consider three types of vital-signs: blood pressure (BP), oxygen saturation (OX) and weight (WT). For MdM, we have two types of monitoring: using an automatic dispenser and using a manual dispenser.

Due to specific requirements of the care-receivers who participated in the field test, we have used combinations of these applications. We involved the care-givers while defining these combinations to make them as realistic as possible.

4.2 Actors

Our field test has been done at Orbis¹, a care-institution in the Netherlands. This institution owns residential blocks where elderly can live and receive care services that are provided by professional care-givers. The aim of this institution is to provide round-the-clock services to their care-receivers and at the same time to enable them to live an independent life as much and as long as possible. The institute provides 8 care-receivers and 3 care-givers as end-users our field test.

There are 3rd-party application services which are used by our homecare applications: calendar, reminder, vital-sign measurement, medicine dispenser and reporting service. The calendar, reminder and vital-sign reporting services are provided by the Biomedical Signals and Systems (BSS) group of the University of Twente². These services are running on Tablet PCs available to the care-givers and care-receivers. The tailoring platform is provided by the Information System (IS) group of the University of Twente³. It is running as an application on Tablet PCs available to the care-givers. If the application logic needs to be updated manually to address unforeseen changes, a programmer of IS modifies the application logic and accordingly updates the service plan.

The vital-sign measurement services are provided by the MobiHealth⁴ company. Care-receivers use a vital-sign measurement device, which is connected to a server in MobiHealth. The MobiHealth server forwards vital-sign measurement values (e.g., blood pressure), which it receives from the measurement devices, to the DHSP platform. We also have an alert service as an internal application of the provisioning platform. The alert service sends an alert to a care-giver's PDA when there is a hazard situation.

The automatic medicine dispenser is provided by the Innospense⁵ company. Care-receivers use the automatic medicine dispenser, which is connected to a server in Innospense. The Innospense server forwards the medicine intake information (e.g., time stamps) to the DHSP platform. For the manual medicine dispenser, we use a simple box in combination with the reminder and alert services.

¹ <http://www.orbisconcern.nl/>

² <http://www.utwente.nl/ewi/bss/>

³ <http://www.utwente.nl/ewi/is/>

⁴ <http://www.mobihealth.com/>

⁵ <http://www.innospense.com>

4.3 Measurement Instruments

To collect data, we logged all the interactions among the DHSP platform and application services. The system logs have the timestamp and the data of the interactions. They are stored in a SQL database. At the end of each experiment, we analyze them off-line. To evaluate the evolvability, during the first experiment, we maintain a list of changes, which are requested by the care-givers. These changes have been applied after the first experiment. In the second experiment, we investigate how effective the applied changes are. Furthermore, after each series of the experiments, we have interviewed the care-givers and care-receivers who participated in the experiments using questionnaires. The first interview was about the perceived effectiveness and efficiency and the requested changes by the care-givers to improve the system for the second experiment. The second interview was about the perceived effectiveness and efficiency of the system in the second experiment and also some general aspects such as pros and cons of using the proposed system.

5 Results of the First Experiment

The applications used in the first experiment are: VsM (BP, WT, OX), Mdm using a manual medicine dispenser and SaM. We explain the result of the first experiment according to the three types of dynamicity: *adaptivity*, *tailorability* and *evolvability*, both objectively and subjectively.

5.1 Adaptivity

We group the system tasks into four categories: sending reminders, sending alerts, vital-signs measurements and medications dispensing. For reminder and alert, the functionality consists of sending the reminder/alert messages and receiving the acknowledge from the application service. For vital-sign and medication, the functionality consists of receiving the vital-sign/medication intake data, sending the data to the reporting service and sending back the acknowledge to the vital-sign measurement/medicine dispenser service. Table 1 shows how effective and efficient the application functionalities are adapted in the first experiment according to the system logs. For instance, during the first experiment (which lasted 2 months), with 8 care-receivers using the DHSP platform, the reminder task was in total 339 times executed, and 46 out of the 339 times the execution was not successful; and the duration of a single execution of this task was between 3419 ms and 10974 ms with an average of 6426 ms and standard deviation of 664.

For effectivity, the unsuccessful numbers of activities are calculated based on receiving exception errors or based on feedbacks from the care-givers. There are four reasons for the unsuccessful tasks: 1) operating system update: the DHSP platform was running on a Windows 2008 server. The operating system updates itself every day at 3 AM by default. 2) application update: during the experiment,

Table 1. Effectivity and efficiency of adapting the functionalities of the applications

	Effectiveness			Efficiency (millisecond)			
	#Total	#Unsuccessful	#Success rate	#Min	#Max	#Average	#Sdv
Reminder	339	46	%86.4	3419	10974	6426	664
Alert	171	16	%90.6	180	1379	356	176
Vital-sign	247	21	%91.4	6255	6712	6392	97
Medication	55	12	%78.1	165	11888	922	2217

one of the service providers updates its application services without informing the platform providers. 3) behavioral change of the care-receiver: after introducing the system, some care-receivers measure their vital-signs much earlier than expected, for instance at 5 AM instead of 8 AM, which is the scheduled time. Then if a care-receiver's vital-sign values are too high/low, the application must send the alert immediately after 5 AM. But in the first experiment, the application starts only half an hour before the scheduled time to check the measured values. Thus some alert messages were not sent on time. 4) duplicated vital-sign measurement values: due to the Bluetooth network used by MobiHealth, sometimes a vital-sign value was sent twice to the platform and the platform forwards the value to the reporting service two times accordingly.

For efficiency, the system measures the time from sending (or receiving) data until receiving (or sending) an acknowledge. For instance, the time of vital-sign task is calculated from receiving a vital-sign value until sending back an acknowledge to the vital-sign measurement service.

Based on the results from our first interview, we conclude that care-givers are not satisfied with the effectiveness of the system. Specially missing alerts are considered unacceptable in life-threatening situations. Besides, when care-receivers cannot use the system for several times, they are not interested in technical reasons and they may lose their trust and interest in using the system. However, the efficiency was considered acceptable. The care-givers mention that the alerts must be delivered *immediately* after the vital-signs measurements in case the values are higher/lower than a predefined threshold. We tried to quantify that and we found out that less than one minute would be considered as *immediately* by the care-givers. Looking at the Table 1, we can see that the efficiency of successful tasks are acceptable for the care-givers.

5.2 Tailorability

In the first experiment, care-givers can tailor/create three types of service plans: VsM, MdM (manual) and SaM. The tailorability task consists of receiving a new service plan by the DHSP platform, deploying the service plan to application services and sending back the acknowledge to the tailoring platform. This is done on the fly without interrupting running applications. Table 2 shows how effective and efficient is the tailorability of the applications. For instance, during the first experiment, VsM application was tailored in total 134 times, and 12 out of the 134 times the tailorability task was not successful; and the duration of a single

execution of this tailorability task was between 296 ms and 85068 ms with an average of 5252 ms and standard deviation of 13202 ms. Some tailorability tasks were unsuccessful due to duplicated primary key error. This error happened because the tailoring platform enables care-givers to delete an existing service plan by deleting its events from the calendar service and adding a new service plan and its corresponding events. The calendar service used the Ids (i.e., identifier) of deleted events for the new events, while the tailoring platform used these Ids as primary key and does not delete them permanently to keep the activity logs of tailoring tasks. Therefore, for some tailorability tasks, the process was interrupted by the duplicated primary key error.

Table 2. Effectivity and efficiency of tailorability tasks

	effectiveness			Efficiency (millisecond)			
	#Total	#Unsuccessful	#Success rate	#Min	#Max	#Average	#Sdv
VsM	134	12	%91	296	85068	5252	13202
MdD	30	5	%83.3	316	27337	3069	6458
SaM	270	15	%94.4	16425	97654	26780	19876

For efficiency, we measure the time from receiving a service plan until sending back the acknowledge to the tailoring platform. Based on the results from our first interview, we conclude that care-givers are satisfied with both effectiveness and efficiency of the tailorability. The number of unsuccessful tailorability tasks was tolerable by care-givers since there was no effect on the running applications.

5.3 Evolvability

During the first experiment and also our first interview, we collected the changes which are requested by care-givers (i.e., unforeseen changes). Some of the changes are related to end-user interfaces of application services for instance, showing less text in the calendar service. Some other changes should be addressed on the DHSP platform and thus we investigate how the platform supports the evolvability of the applications. These changes are listed as follows: 1) If care-receivers measure their vital-signs earlier than the scheduled time and the measured values are not in the normal range, the alert must be sent immediately. 2) The vital-sign values should be sent with the alert messages to help care-givers to be prepared in advance. 3) For weight measurement, first we sent an alert when the weight of a care-receiver was either higher or lower than the last measured weight more than a threshold. Later on, care-givers want to receive this alert if the weight of a care-receiver is either higher or lower than a threshold. 4) The nurses can add more than one event per day to the calendar service for each application.

The programmer modifies the application logic manually to evolve the applications based on the unforeseen changes. To address the unforeseen change 1, we add an alert activity at another orchestration pattern that receives vital-sign

values from the MobiHealth. The alert activity calls the decision service of VsM application to see whether it is necessary to send alert or not. Since, we have reused the VsM decision service in another orchestration with the same service interface, the modification is accomplished quickly by dragging and dropping an alert activity to the orchestration. To address the unforeseen changes 2, 3 and 4, the modification is required only in the decision service without changing the orchestrations.

We improved evolvability of the applications because of using the decision service. However, using the decision service increased time and data communication to run the applications. In the first experiment, the decision service has been called 256 times. It takes 278 milliseconds by average (min=149,max=1700, Std=176 milliseconds) to call the decision service. This period of time is much less than the time durations of the system tasks (see Table 1) and is not noticeable by care-receivers and care-givers. The data model of messages between the orchestration and the decision service consists of 19 variables (8 String, 7 Integer and 4 Boolean variables). In our field test, the size of data which is exchanged between an instance of orchestration and the decision service is always less than 5 kilobytes for each interaction. Therefore, we have seen that using of the decision service improves evolvability while its cost in terms of time and data communication is rather trivial.

6 Results of the Second Experiment

Two months after the first experiment, the improved version of the system based on the requested changes was validated in the second experiment. The applications used in the second experiment are: VsM (BP, WT), MD (using manual and automatic dispenser) and SaM. In the second experiment, we empowered the system with an extra power supply, disabled automatic operating system update and asked all the service providers not to modify their application services during the experiment. We first evaluate the adaptivity and tailorability of the system to see if the system is improved. After execution of the second experiment, we interviewed the care-givers to measure their perceived *effectiveness and efficiency*. Moreover, we asked the care-givers to give us their opinions about the whole system through open-end (descriptive) questions. Even though the focus of this work is the DHSP platform and its implemented prototype, some of the results reported in this section are in general about the whole system.

6.1 Adaptivity

Based on the result, the effectiveness was improved since we had only four unsuccessful vital-sign tasks. The reason was that the reporting service took longer than the first experiment. Therefore, the DHSP platform sent an acknowledge to the vital-sign measurement service with a delay and it caused an error on care-receivers' PDAs. Although the vital-sign values are received and shown correctly, we consider them as unsuccessful task since showing this error on the PDA was

inconvenient for care-receivers. The achieved efficiency of the second experiment is almost the same as the one in the first experiment. Based on our second interview, the care-givers are satisfied with both effectiveness and efficiency of the system tasks' adaptivity.

6.2 Tailorability

We have achieved 100% effectiveness regarding VsM and MdM tailorability. However, we had several failure for SaM tailorability because care-givers create more than 5000 social events per each service plan deployment. Thus, the deployment took more than 3 minutes and they stopped using the application since they thought it was broken. Based on our second interview, care-givers are satisfied with the effectiveness and efficiency of the VsM and MdM tailorability, but not with the SaM tailorability.

6.3 General Feedback

We experienced that not all the care-givers could distinguish the different parts of the system. Therefore, our second interview contains 20 task-independent questions to ask care-givers' opinions regarding the whole system as an integrated application. These are the lessons learned:

- Care-givers believe that the proposed homecare system can work in practice particularly after the second experiment. Note that the system is not used occasionally, but in daily use with more than 400,000 transactions among the services. However, the end-user interfaces for care-receivers must be improved (e.g., showing less text and bigger icons).
- The system would be more useful for private apartments outside of the care-center because it can save lots of traveling time for care-givers. Inside the care-center, it is sometimes faster to reach the care-receiver instead of using the system.
- In close future, care-receivers with less health problems will be advised by the government to stay at home in order to reduce healthcare costs and thus, the system would be even more promising.
- The system reduces physical contact that can decrease care quality. In this case, a voice/video communication can be helpful.
- Although adding vital-signs values to alert messages was useful, it is not sufficient to decide what they should do before visiting care-receivers. So, a video communication can help care-givers to judge.
- It is highly desirable that care-receivers measure their vital-signs without waiting for a care-giver particularly when they want to leave their apartments. However, care-receivers would be obliged to get back home, if they were outside their apartments at the scheduled time. Therefore, it is useful if they can carry the measurement devices with themselves.
- The field test can be improved if it takes longer than 4 months with better target group of care-receivers (e.g., 70-80 years old).

- An automatic dispenser is more successful than the manual one because first, it has an embedded reminder beep and pressing a button on the dispenser device is easier than pressing a message box on Tablet PC, and second, it cuts the bag that contains the medicine.

7 Conclusion

In this work, we discussed a field test of our dynamic homecare service provisioning (DHSP) platform. We investigated whether the following goals can be supported by the platform: (a) To adapt a homecare application successfully within a certain time expected by an end-user, (b) To deploy a (re)tailored homecare application successfully within a certain time expected by a care-giver, and (c) To deploy an evolved homecare application successfully within a certain time expected by a programmer. At least one of the above goals should be achieved in order to justify using the proposed DHSP platform. With respect to these goals, we investigated the usability, i.e., efficiency, effectiveness and satisfaction (perceived efficiency and effectiveness) of the homecare applications running on the platform.

We believe that dynamic homecare applications provisioning must be efficient for the platform, care-giver, and programmer, i.e., the required time for adapting, (re)tailoring and evolving the applications with respect to the contextual changes must be as less as possible. Our interviews indicated that a desirable property of the DHSP platform is to decrease the work load of the care-givers, thus saving costs for the care centers and care-provider organizations. In addition, saving cost on manpower (needed for manual tasks) should not be annihilated by extra costs for system resources and application programmers. This cost saving for addressing new contextual changes is very important in the homecare domain since its environment is often subject to several types of changes. Our field test shows that the efficiency of adaptivity is acceptable for the care-giver although the response time is higher than that of a stand-alone application due to distributed application services. The efficiency of tailorability is also acceptable for the care-givers except for the SaM application due to the large numbers deployed events in each tailoring task. Separating the decision making rules from the orchestrations improves the evolvability of the applications since almost all the changes have been handled within the decision rules without redeploying the orchestration patterns.

Regarding the effectiveness, the first experiment was not successful due to several system failures. These failures caused several safety and availability risks using our proposed platform. However, it motivated us to develop a risks identification method which we applied after the first experiment to prevent similar risks in the second experiment. As a result, the effectiveness of the homecare applications in the second experiment is acceptable for the care-givers.

The care-givers believe that having a DHSP platform enables them to use alternative application services (e.g., manual or automatic medicine dispenser) for a specific homecare application (e.g., MdM application) without affecting how the care-givers tailor and how the care-receivers use that homecare application.

This would be useful to decrease the required time from both care-givers and care-receivers to learn how to use a new homecare application. Moreover, using the DHSP platform enables the care-giver to create applications with more realistic and useful functionalities. For instance, in our field test, the care-givers created a new application to ask a care-receiver to take a medication if his blood pressure is in a specific range, which can be done by integrating a medicine dispenser and blood pressure measurement devices.

Our conclusion from the field test is that the DHSP platform is usable (by care-givers and care-receivers) at least in our field test. However, the number of homecare applications and end-users involved in our field test is limited. To be able to provide statistical analysis, we plan to provision more homecare applications with more care-givers and care-receivers as our future work. In addition, our field test shows that both care-givers and care-receiver are interested to see more often measured vital-sign values. But it is important (1) how to show this data to the end-users (e.g., using graphical interface or statistical analysis) and (2) to export the data automatically to other healthcare applications (e.g., hospital patient record). As such, using application services with data processing user-interfaces and integrating the vital-sign measurement and medication intake data with other existing healthcare information systems are considered in our future plan.

Acknowledgements. This work is part of the IOP GenCom U-Care project (<http://ucare.ewi.utwente.nl>) which is sponsored by the Dutch Ministry of Economic Affairs under contract IGC0816.

References

1. Batet, M., Isern, D., Marin, L., Martinez, S., Moreno, A., Sanchez, D., Valls, A., Gibert, K.: Knowledge-driven delivery of home care services. *Journal of Intelligent Information Systems*, 1–36 (2010)
2. Bevan, N., Kirakowski, J., Maissel, J.: What is Usability? In: *Human Aspects in Computing: Design and Use of Interactive Systems with Terminals*, pp. 651–655. Elsevier (1991)
3. Breivold, H., Crnkovic, I., Eriksson, P.: Evaluating software evolvability. *Software Engineering Research and Practice in Sweden*, 96 (2007)
4. European Commission: Ageing well in the information society - an i2010 initiative - action plan on information and communication technologies and ageing. Tech. rep., EU (June 2007)
5. Fraile, J.A., Bajo, J., Abraham, A., Corchado, J.M.: HoCaMA: Home Care Hybrid Multiagent Architecture. In: *Pervasive Computing. Computer Communications and Networks*, pp. 259–285. Springer, London (2010)
6. Frøkjær, E., Hertzum, M., Hornbæk, K.: Measuring usability: are effectiveness, efficiency, and satisfaction really correlated? In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 345–352. ACM (2000)
7. Gabner, K., Conrad, M.: ICT enabled independent living for elderly, A status-quo analysis on products and the research landscape in the field of Ambient Assisted Living in EU-27. prepared by VDI/VDE Innovation und Technik GmbH (March 2010)

8. ISO: Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability. International Standard 9241-11 (1998)
9. Kleinberger, T., Becker, M., Ras, E., Holzinger, A., Müller, P.: Ambient intelligence in assisted living: enable elderly people to handle future interfaces. In: *Universal Access in Human-Computer Interaction. Ambient Interaction*, pp. 103–112. Springer (2007)
10. Malanowski, N., Ozcivelek, R., Cabrera, M.: *Active Ageing and Independent Living Services, The Role of Information and Communication Technology*. European Community (2008), <http://www.unic.pt/images/stories/publicacoes2/JRC41496.pdf>
11. MATCH: Mobilising Advanced Technologies for Care at Home (2005-2012), <http://www.match-project.org.uk/main/main.html> (last visited: September 2012)
12. McGee-Lennon, M.R.: Requirements engineering for home care technology. In: *26th Annual SIGCHI Conf. on Human Factors in Computing Systems*, pp. 1439–1442 (2008)
13. MPOWER: Middleware Platform for eMPOWERing cognitive disabled and elderly (2006-2009), <http://www.sintef.no/Projectweb/MPOWER>
14. Rao, J., Su, X.: A Survey of Automated Web Service Composition Methods. In: Cardoso, J., Sheth, A.P. (eds.) *SWSWPC 2004*. LNCS, vol. 3387, pp. 43–54. Springer, Heidelberg (2005)
15. Shackel, B.: The concept of usability. In: *Visual Display Terminals: Usability Issues and Health Concerns*, pp. 45–87. Prentice-Hall, Englewood Cliffs (1984)
16. Urbietta, A., Barrutieta, G., Parra, J., Uribarren, A.: A survey of dynamic service composition approaches for ambient systems. In: *Proceedings of the 2008 Ambi-Sys Workshop on Software Organisation and MonIToring of Ambient Systems, SOMITAS 2008*, pp. 1:1–1:8. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), Brussels (2008)
17. Wieringa, R.: *Design Methods for Reactive Systems: Yourdon, Statemate, and the UML*. Morgan Kaufmann (2003)
18. Wieringa, R.: A unified checklist for observational and experimental research in software engineering (version 1) (March 2012), <http://doc.utwente.nl/79890/>
19. Wieringa, R., Morali, A.: Technical Action Research as a Validation Method in Information Systems Design Science. In: Peffers, K., Rothenberger, M., Kuechler, B. (eds.) *DESRIST 2012*. LNCS, vol. 7286, pp. 220–238. Springer, Heidelberg (2012)
20. Zarghami, A., Sapkota, B., Zarifi Eslami, M., van Sinderen, M.: Decision as a Service: Separating Decision-making from Application Process Logic. In: *The 16h IEEE Int.l Enterprise Distributed Object Computing Conference (EDOC)*, pp. 103–112. IEEE (2012)
21. Zarghami, A., Zarifi Eslami, M., Sapkota, B., van Sinderen, M.: Dynamic Homecare Service Provisioning Architecture. In: *9th Int. Conf. on Service-Oriented Computing and Applications (SOCA)*, pp. 292–299 (2011)
22. Zarghami, A., Zarifi Eslami, M., Sapkota, B., van Sinderen, M.: Service Realization and Compositions Issues in the Homecare Domain. In: *6th International Conference on Software and Data Technologies (ICSOFIT)*, vol. 1, pp. 347–356. SciTePress (July 2011)
23. Zarghami, A., Zarifi Eslami, M., Sapkota, B., van Sinderen, M.: Toward Dynamic Service Provisioning in the Homecare Domain. In: *5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth), Workshop on Designing and Integrating Independent Living Technology*, pp. 292–299. IEEE (May 2011)