End-User Development of Activity-Supporting Services for Smart Homes

Rafik Belloum

End-user development of activity-supporting services for smart homes

Rafik Belloum

The University of Bordeaux / Inria

Bordeaux, France

rafik.belloum@inria.fr

Supervised by Charles Consel

Abstract—**Independent living often requires assistive computing support to be tailored to the specific needs of each older adult. We propose a tool-based approach to allowing caregivers to define services for in-home daily living activities, leveraging their knowledge and expertise on the older adult they care for. This approach consists of two stages: 1) a wizard allows caregivers to define an assistive service, which supports aspects of a daily activity, specific to an older adult; 2) the wizard-generated service is uploaded in an existing smart home platform and interpreted by a dedicated component, carrying out the caregiver-defined service.**

Index Terms—**Smart home, Ambient assisted living, Assistive computing, End-user programming**

I. Introduction

Supporting older people in performing their everyday activities is a new avenue of research in the field of smart homes. In recent years, pervasive computing systems have been developed to provide assistance for older adults' to age-in-place. It has been shown that following a human-centred approach by customizing assistive services in a smart home to the user abilities plays a key role towards achieving effectiveness and acceptance of theses systems [1]. The human-centered approaches have been used in real-world experiments for supporting activities of daily living (ADLs) and have shown numerous benefits, including reducing the burden on caregivers and preventing any gap between the behaviour of services they deliver and the needs expressed by the user and/or their caregivers (*e.g.,* [1]).

Even though satisfying for users, the human-centred approach is a time-intensive exercise. In fact, from a software engineering point of view this approach presents a significant challenge to the development of activity-supporting services by preventing to implement a generic service per daily activity since it is necessary to implement a multitude of services, tailored for each person and their specific environment. For instance, a service detecting a daily activity such as meal preparation and providing assistance to support this activity has to (1) closely match the way each person performs this activity, which may involve using various appliances within customizable time slots; and (2) providing the type and level of assistance adapted to each person's needs.

To resolve this software development bottleneck, we set out to analyze exisiting activity-supporting services. This analysis revealed that they had extensive common properties, which suggested that they formed a program family. As reported in the literature, the variations and commonalities of a program family can be leveraged to factorize parts of the software development process. Factorization typically takes the form of domain-specific languages and gives rise to program generation tools. This finding was a key insight toward solving our software development bottleneck problem.

II. Our approach

We propose a complete approach to developing activity-supporting services, ranging from the modeling of the target activities, to an end-user tool to define services, to a layer to run services in a smart home.

A. Taxonomy of activities for independent living

to model the activities to be supported by our approach, we have organized the commonalities and variabilities identified into a taxonomy, which classifies technology-supported activities and elicits their characteristics. As such, it can serve as a guide for caregivers to define an activity-supporting service and address the user's needs. At the root of this taxonomy, an activity needs to have a description. It may be recurring and may be supported by actions. The next level introduces a choice between outdoor, indoor, and home visit-related activities. Outdoor activities require a date and a time at which the user is notified to start preparing, as well as a date

and a time at which the user is supposed to have departed from home. User departure can be checked via a detector. Indoor activities consist of a date, a time period, and a sensor, if supervision is needed. Finally, home visit-related activities require a date and a time. The leaves of our taxonomy consist of actual activities that inherits the characteristics of the parent levels. Our taxonomy of technology-supported activities suggests a staged process to define services that could be tooled. This opportunity is explored in the next sub-section.

B. A wizard for activity-supporting services

Our aim is to create a tool that 1) covers the taxonomy for technology-supported activities and 2) provides an accessible user interface such that caregivers without programming skills can define services that address their carereceiver's needs. As suggested by our taxonomy of technology-supported activities, defining assistive services should be a staged process, allowing the user to specify the characteristics of the target service in a stepwise manner. To match this requirement, our tool has been designed as a wizard (see Figure 1), which makes explicit the decomposition of a service definition, reducing the risk of errors. We iterated the design of our wizard with caregivers to ensure the activity characteristics were prompted in an order that matched their preference.
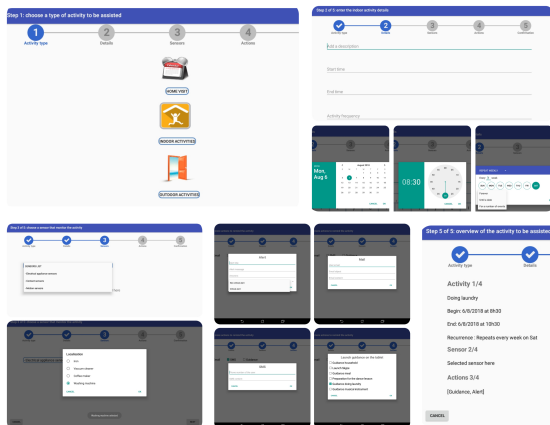

Fig. 1. A wizard for caregiver development of services

C. Execution support for wizard-defined services

Our wizard generates a representation of a caregiver-defined service that is passed to the smart home platform. A dedicated layer is in charge of interpreting this representation to realize the service, leveraging available connected objects (*e.g.,* sensor),

services (*e.g.,* calendar), and interaction modalities (*e.g.,* user notifications).

To validate, our approach we used our wizard to define 43 existing activity-supporting services and observing their behavior equivalence when deployed in the home of our participants. Also, the wizard was used to define a number of new services to test the coverage of the target taxonomy of activities. The wizard-defined services targeted older adults, users with intellectual disability, and users with autism.

III. OUTLINE OF FUTURE WORK

In the future, we plan to extend the kind of sensors that can be used in the wizard. In particular, we would like to introduce a high-level notion of sensors that would allow caregivers to exploit activities that may involve a sequence of sensor activations, some duration, *etc.* Needs for such high-level sensors come from caregivers as they express services they want to define to support independent living of individuals they care for. These high-level sensors should be made available in the wizard, together with simple ones, increasing the coverage of activity-supporting services. Our approach has been implemented using HomeAssist platform. To assess the applicability of our approach, we plan to implement our approach on another smart home platform. We also plan to conduct a usability study involving professionals in aging to evaluate the ergonomic aspect of our wizard.

IV. CONCLUSION

We have characterized an area of home activities that is needed for independent living and can be supported by smart homes. To address this area, we have introduced a wizard-based approach toward empowering caregivers to develop activity-supporting services, leveraging smart home functionalities. As such, our approach allows the expertise of caregivers to be directly applied to defining assistive support for an individual they care for. We have evaluated wizard-defined services by comparing them to manually-programmed services and ensuring that they both had the same behavior.

REFERENCES

[1] Charles Consel. Assistive computing: A human-centered approach to developing computing support for cognition. In *2018 IEEE/ACM 40th International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*, pages 23–32. IEEE, 2018.