Service Reconfiguration in the DANAH Assistive System

Said Lankri, Pascal Berruet, Jean-Luc Philippe

Service Reconfiguration in the DANAH Assistive System

Saïd LANKRI, Pascal BERRUET and Jean-Luc PHILIPPE

European University of Brittany
Lab-STICC, F-56321 Lorient Cedex, France
{said.lankri,pascal.berruet,jean-luc.philippe}@univ-ubs.fr

Abstract. Smart Homes are pervasive systems that interact with the user using a service offer paradigm to provide fully automated daily repetitive tasks. When services are augmented with semantic relationships, one can build adaptive services and systems. In this paper we deal with service failures and propose a recovering method, which we call service reconfiguration, to ensure service availability in smart homes. Both off-line and on-line reconfigurations are considered. This method has been implemented in the DANAH assistive system.

1 Introduction

Assistive technology systems (ATS) are a kind of pervasive systems that help dependant people improve their lives. The Smart Home concept [4] emphasises on environmental control by incorporating electronically controllable devices and sensors to provide automated services and monitoring.
The DANAH assistive system is a software application that helps the elderly and the disabled turn their living spaces to entities that provide assistive services. Within automated homes or medical structures, users can benefit from daily automated tasks using a human machine interface simply by selecting the service they want, and DANAH shall deliver it for them.
In the context of these intelligent environments, we aim at ensuring service availability in the presence of failures through *service reconfiguration*.
This paper is structured as follows : Section 2 introduces service reconfiguration. Section 3 presents the DANAH assistive system. Section 4 presents addressed service failures and the reconfiguration process. Finally, we conclude our work and present eventual perspectives and future work.

2 The DANAH ATS

2.1 Architecture

The DANAH Assistive System [1] is a modular distributed assistive system that allows the disabled and the elderly to interact with their environment using a service offer paradigm, achieving *environmental control*. It relies on three entities : Servers, Clients and Automation Technology as depicted in Fig.1. *Servers* and deployed in the environment and contain information about the topology and the services they locally offer. A *Client* establishes a connection to a server and gets a list of the available services (pull-mode).

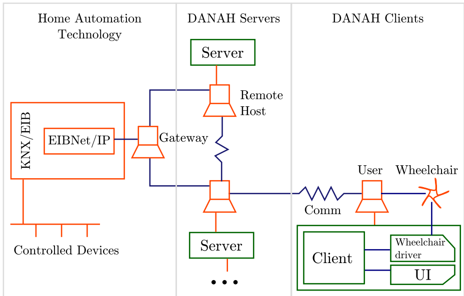


Fig. 1. DANAH architecture and HW/SW mapping, using the KNX/EIB Home Automation Technology

2.2 Resources, Operations and Effects

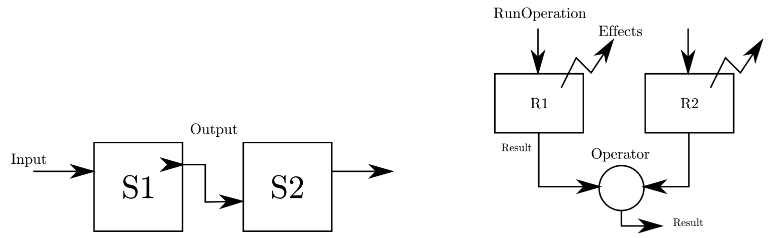
Resources are DANAH representation of controlled home hardware, such as doors, lights and TVs. A resource is advertised using its *name* and has a status stored in its *properties* which are key-value pairs. It contains *operations* that represent the services it provides (e.g. 'On' and 'Off'). A *runtime* is responsible of executing operations and updating resource status. Communication between the runtime and the physical device is achieved using a *protocol*. Resources are linked to the environment topology through *activation nodes* that specify at which areas operations can be delivered. Finally, resources are *tagged* by keywords that inform about their characteristics.
Operations can be *runtime controlled* (atomic) or *composite*. Atomic operations are directly run by calling appropriate method within the runtime. Composite operations are operations described using literal expressions, as shown in Fig.2

```
operation "Enter" { expression="SEQ(Door.Open Light.On)" }
```

Fig. 2. Composite Operation 'Enter' uses two atomic operations

Unlike in traditional service oriented applications, invoking a resource operation does not require input data. Instead, *effects* on other resources are produced, as depicted in Fig.3.

The *effect* [2] [3] of an operation is what is perceived by the user. Its principle is that running a resource operation can have effects on one ore more resources. Effect computation is performed using resource runtimes and *effect rules*. The runtime is responsible of computing the effects of the resource's own operations. Then using effect rules, DANAH computes the overall change on other resources. An effect rule consists of a *precondition* that if satisfied triggers its corresponding *postcondition*, as shown in Fig.4.



(a) Standard service mechanism (b) Service mechanism DANAH

Fig. 3. Composition schemes in traditional applications and in DANAH

```
effect {  pre="MainLight@Status==On"
          post="this@Illuminance=100"}
```

Fig. 4. An effect rule

3 Service Reconfiguration

3.1 From the Request to the Provision

A *service* can be an operation, a composition of operations or an objective to reach (e.g. `Room@Temperature=25`). When it is requested, DANAH starts by performing a *syntactic expansion* which recursively replaces composite operations by their corresponding expressions, and dynamically resolving objectives into a set of operations. To run an operation, DANAH performs navigation to the resource by choosing one of its activation nodes (if any) and finally activates it with the help of the automation technology. Whan an activation fails, it triggers service reconfiguration. Reconfiguration can be either static using off-line rules or dynamically computed at runtime.

3.2 Static Reconfiguration

Static reconfiguration is triggered on activation failure. It uses static rules defined in resource descriptions. When the activation of an operation fails, DANAH searches in all resource descriptions for an alternative expression that may replace the defunct operation, as shown in Fig.5. It says that when the turning on the main light fails, the system must fall back to turning on the two alternative lights.

```
reconfiguration { fail="MainLight.On"
                  alt="SEQ(AltLight1.On AltLight2.On)" }
```

Fig. 5. Static reconfiguration rule

Static reconfiguration rules are convenient ways for the user to set its own reconfiguration preferences, since it has priority on any dynamically computed alternative.

3.3 Effect-based Reconfiguration

Effect-based reconfiguration is triggered on activation failure, in the absence of static reconfiguration rules. The aim of effect-compensation procedure is to compensate the effect the failed operation should have produced on other resources

using one or several other operations from different resources.
When an operation is run, the resource's properties are first updated by its runtime. Then using effect rules, these property changes may satisfy preconditions that trigger other changes in other resources, as depicted in Fig.6

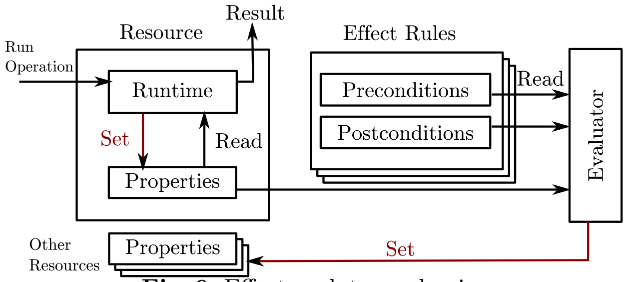


Fig. 6. Effect update mechanism

When an activation fails, effects of the failed operation are searched and compensation is performed for each affected resource. This is done using a 'dry run' mechanism that selects the operations which change the considered effect and fakely running them. At the end, the sequence that produces the closest effect that the one that is expected is suggested as an alternative.

4 Conclusion

In this paper we described the service reconfiguration in the DANAH assistive system, which aims at ensuring service availability in case of failures. Addressed failures are operation activations. Static reconfiguration is used on activation failure and relies on off-line reconfiguration rules. Effect-based resource linking has been introduced and used to perform effect compensation, possibly using several operations to compensate the effect of a single one.

References

1. Saïd Lankri, Pascal Berruet, André Rossi, and Jean-Luc Philippe. Architecture and models of the DANAH assistive system. In *SIPE '08: Proceedings of the 3rd international workshop on Services integration in pervasive environments*, page 19–24, New York, NY, USA, 2008. ACM.
2. Aitor Urbieta, Ekain Azketa, Inma Gomez, Jorge Parra, and Nestor Arana. Analysis of effects- and preconditions-based service representation in ubiquitous computing environments. *icsc*, page 378–385, 2008.
3. Aitor Urbieta, Ekain Azketa, Inma Gomez, Jorge Parra, and Nestor Arana. Towards effects-based service description and integration in pervasive environments. In *SIPE '08: Proceedings of the 3rd international workshop on Services integration in pervasive environments*, page 1–6, New York, NY, USA, 2008. ACM.
4. V. Ricquebourg, D. Menga, D. Durand, B. Marhic, L. Delahoche, and C. Loge. The smart home concept : our immediate future. In *Proc. IEEE International Conference on E-Learning in Industrial Electronics*, page 23–28, December 2006.