Explaining The Operation of A Home Care System

Kenneth J. Turner,
Computing Science and Mathematics, University of Stirling,
Stirling FK9 4LA, UK Email: kjt@cs.stir.ac.uk

23rd April 2015

1 Introduction

The need for home care technologies is introduced. Rule-based management offers flexibility and customisation of a home care system. Related work is discussed in home care technologies, policy-based management and expert systems.

1.1 Need for Home Care Technologies

The world population is ageing, with the percentage of older people (over 65) gradually rising. In the UK, for example, this percentage was 24.4% in 2000 and is expected to be 39.2% by 2050 (Select Committee on Economic Affairs 2003). A similar situation applies in other developed countries, with much higher percentages forecast for some areas (e.g. 71.3% by 2050 in Japan). Clearly this will increase the demand for care of older people. Although people are living for longer, many have to deal with long-term, age-related conditions. The growing percentage of older people, coupled with pressure on social and health care budgets, means that care providers will be increasingly challenged to cope. As a result, it will not be feasible to provide sufficient care homes (which are much more expensive than looking after someone in their own home).

Technology to support home care delivery has been identified as part of the solution. Telecare (also called assisted living) refers to remote support of social care at home. This includes monitoring for undesirable situations (e.g. falls, flooding, night wandering) as well as services for the less able (e.g. curtain openers, door entry phones, home automation). Telehealth refers to remote support of health care at home. This includes remote consultation and diagnosis as well as monitoring health parameters (e.g. blood pressure, heart rate, seizure risk).

There is a trend towards more integrated and holistic care. As an example, the Single Shared Assessment performed in Scotland allows social and health care needs to be evaluated (Scottish Executive 2001). As a result of a care assessment, technology-supported solutions for both may now be prescribed. There are also possibilities for commonality in facilities to support social and health care. The boundary between social and health care has thus become blurred, with changes in thinking about telecare and telehealth solutions. This can be seen, for example, in the shift away from purely telecare or telehealth towards independent living technologies and life-enhancing technologies. These are umbrella terms encompassing a blend of telecare, telehealth, and mainstream technologies that support wellbeing and health.

Home care systems are computer-based systems that support delivery of care. Typically, some kind of home hub is provided to collect, analyse, react to, and forward care data collected from a variety of sensors or other input devices. Besides sensors for inputs, a home care system can use software services (e.g. for communication, speech input-output, or weather forecasts). A home care system is able to respond through a variety of actuators to control appliances, maintain the home environment, signal alert conditions, etc. More sophisticated systems have a degree of programmability, allowing customisation for individual user needs and adaptation to changing circumstances.

Telecare aims to provide computer-based support for daily activities. At the minimum, this involves monitoring the extent to which people are living normally at home. For example, non-intrusive sensing can confirm that the individual is sleeping well, is active around the house, and is dealing with personal hygiene and toileting. A telecare system will also typically check for potentially hazardous situations such as a gas cooker not being lit, water being left running, or the user falling. More advanced systems can help with activities through speech-based or visual prompting, and by reminders such as for medication and appointments. Trends and anomalies can be noticed in user behaviour and reported to carers or to the users themselves for further investigation (Munstermann, et al. 2012).

Home care technologies offer significant benefits. Particularly in rural settings, the ability to support care at a distance can save substantial travel. Many health authorities are promoting self-care at home rather than relying exclusively on centrally provided care. Trends, anomalies and alert conditions can be identified and reported to a central location (e.g. a health centre or a call centre). Family members can be reassured that the user is getting on normally. Professional carers can also be relieved of low-level monitoring tasks. Older people can therefore be assisted to stay longer in their own homes, where they are in familiar surroundings and near to the people and the area they know.

Recent work has addressed the need to empower end users by providing methods for people to monitor and visualise their own activities and data. This allows people to make proactive and preventative care and lifestyle decisions for themselves. Older people can therefore be assisted to stay longer in their own homes, where they are in familiar surroundings and near to the people and the area they know.

1.2 Rule-Based Management of Home Care

The work discussed in this paper is a response to the need for home care technologies. However, in general the aim has been to support flexible and automated management for systems of whatever kind. The idea is to expose the system logic and to make it available for easy change. When it comes to adaptation and customisation of the system, this largely removes the need for specialised technical knowledge and programming ability.

The work uses rules (goals and policies) to determine how the system should react to inputs. This is a generic approach that has been used by the author for management of care delivery in the home, for sensor networks in wind farms, and for call control in (Internet) telephony. For concreteness the paper is illustrated through application to home care as an important and topical field of study.

Commercial home care systems are relatively fixed in their functionality. Even where modification is possible, this usually requires specialised expertise and perhaps reprogramming. In contrast, the work described here allows users to define how the system should react to occurrences around the home. Thus an end user might state what to do in the event of a fall (e.g. send an alert message), or a doctor might define an appropriate response to forgetting medication (e.g. issue a spoken reminder).

The automation and flexibility offered by the approach are positive benefits. However, it had previously been observed that non-technical users could find it difficult to predict what the home care system would do in various circumstances. Users could also find it hard to understand why the system had (or had not) done something. This paper reports a solution to these problems by offering a user-friendly way to check system behaviour, i.e. its past and future actions.

The approach is important for all users of home care technologies: the end users themselves (the residents), their informal carers (family and friends), and their formal carers (health and social care professionals). It allows users to customise and adapt home care technologies for individual and evolving needs. More particularly for this paper, it allows users to gain a better understanding of the home care system and how it has been set up.

The overall goal of this work is to provide a flexible system that can offer support for daily activities in the home. The paper describes a research prototype of a home care system. Although this has been trialled in actual homes, it has not provided core support for vulnerable users. For reasons of safety, mainstream care services have remained as the primary source of support. The prototype is not a commercial solution, but the hope is that it will influence the design of the next-generation commercial home care systems.

An important aim has been to make the system adaptable and customisable for the specific and evolving needs of end users. For users with a degenerative condition (e.g. dementia), the support required is likely to change over time. The end users, or more likely their carers, should be able to adjust the behaviour of the home care system to reflect these changing requirements. One aspect of adaptability addressed in this work is being able to define rules for how the system should react to different circumstances. This requires the system to be able to explain itself so that users can understand its operation. For example, why did the system take a certain action (or not do this)? For *any* automated system, it can be difficult to understand what is being done by the system on behalf of the user. The explanation facility described in this paper has therefore been a necessary and vital addition to the system.

1.3 Related Work

1.3.1 Home Care Systems

Technology for home care has been enthusiastically embraced as part of the solution for the ageing population. Home care systems are computer-based solutions for supporting delivery of care (Turner & Maternaghan 2012). Typically some kind of home hub is provided to collect, analyse, react to, and forward care data from a variety of sensors or other input devices. Besides sensors for inputs, a home care system can use software services (e.g. for communication, speech input-output or weather forecasts). A home care system is able to respond through a variety of actuators to control appliances, to maintain the home environment, to signal alert conditions, etc. More sophisticated systems have a degree of programmability, allowing customisation for individual user needs and adaptation to changing circumstances.

Home care includes monitoring the extent to which people are living normally at home. For example, non-intrusive sensing can confirm that the individual is sleeping well, is active around the house, and is dealing with food preparation and toileting. A home care system will also typically check for potentially hazardous situations such as a gas cooker not being lit, water being left running, or the user falling. More advanced systems can help with activities through speech-based or visual prompting, and by reminders such as for medication and appointments. Trends and anomalies can be noticed in user behaviour and reported for further investigation (Munstermann et al. 2012).

Home care technology is said to have gone through three generations (Doughty, et al. 1996). The first generation of reactive home care systems mainly focused on social alarms. For example, users could use a pendant to signal a call centre for help or to contact support staff. The second generation of proactive home care systems allowed more automatic responses based on sensor information. For example, a fall detector could automatically report an alert condition without the need for user intervention. The third generation of integrated systems is aimed at enhancing the user's quality of life. For example, virtual communities can link the user into a wider care network and can provide access to remote services for communication and advice.

An interesting reflection on assistive environments is provided by (Helal, et al. 2012), based on the Gator Tech Smart House. Several of the requirements identified in that work are echoed in the present paper. For example the need for integration has been met through a consistent interface for services and devices, while programmability and evolution have been achieved through goals and policies. A significant difference is that adaptability and customisability are accomplished through user-defined rules rather than actual programming as in the Gator Tech work.

1.3.2 Policy-Based Management

Computer-based policies have been used in applications such as controlling access to resources (e.g a database), managing devices in a system or network (e.g. a router), and ensuring that networks provide an adequate quality of service (e.g. throughput). Policies are rules that are automatically applied when events occur. Most policy languages are in an ECA form that deals with Events (triggers when something happens, e.g. the user falling), Conditions (restrictions on when a rule applies, e.g. during the day), and Actions (how the system should react, e.g. alerting a carer). Some illustrative examples of policy-based systems, drawn from a large field, are as follows:

- Homer (Maternaghan & Turner 2011) uses rules to manage the home. It is mostly focused on home automation, though it is also relevant to home care. The main contributions of Homer are an extensible component architecture and a policy system that is well-integrated with this.

- Police (Dursun & Örencik 2003) follows a traditional policy-based approach. It deals particularly with conflicts among policies in a distributed setting. For example, a policy that allows any home user to access care data may conflict with a health service policy that this be restricted to medical staff. The approach avoids certain kinds of conflicts, but provides mechanisms for handling domain-specific conflicts.

- Ponder (Damianou, et al. 2001) is a well-known policy-based approach. It offers a mature methodology for handling policies in applications such as system management and sensor networks. Ponder supports zones where policies apply, conflicts among policies, and realisation of goals using lower-level policies.

For the work in this paper, the human aspects of home care tend to rule out the more technically-oriented policy approaches used in system management. As argued in (Turner, et al. 2006), a different kind of policy approach is needed for 'softer' management tasks of the kind found in human-oriented systems like home care.

(Shafti, et al. 2013) use ECA rules for ambient intelligence in a home context. This work is complementary to that reported here as it aims to infer what the rules should be. The approach observes how people interact with their environment and can thus infer what their preferences are, such as for lighting levels.

(Leong, et al. 2009) describe a rule-based system for smart homes. However, this is a rather heavy-weight solution that expects home devices to be interconnected via an Ethernet local area network. The system supports basic ECA rules, but these do not seem to be intended for definition by end users.

Context-aware systems, e.g. (Bardram 2004), aim to make a system reactive to context, and in that sense have some affinity to policy systems. Gaia (Román, et al. 2001) creates 'active spaces' from physical spaces supplemented by a context-aware infrastructure. However, context awareness is a separable aspect. The policy system described in this paper accepts information from an external context system in order to influence its behaviour. Any third-party system could be used to provide this information.

1.3.3 Explanation in Expert Systems

An expert system codifies specialised knowledge in the form of rules that can be applied to given facts in order to draw some conclusions. However, it is often necessary to explain why these conclusions were reached. For example, a medical expert system might be required to explain how it reached a certain diagnosis and recommended treatment (Swartout 1983).

Explanation in expert systems has a long history, dating back almost to their inception. An easy way to provide some kind of explanation is simply to trace back through the rules that were used to derive the conclusions. However, this is more of value in debugging than in giving a coherent explanation of the system's reasoning (Moore & Swartout 1988). Instead, developers have focused on defining a rich representation of the system's knowledge, an explicit understanding of its problem-solving strategies, and a deeper handling of natural language in queries and explanations (Moore & Paris 2007).

Policy-based and rule-based management offer considerable flexibility and the opportunity to customise the system to user needs. However, explaining system actions to the user is something that has received almost no attention. This is probably because such systems are usually designed for use by experts who do not need this. The work in this paper is a response to this lack. Although the new approach is relevant for any domain, it is illustrated for the particular application of home care.

Policy-based systems are significantly different from expert systems. As a result, the way that expert systems can explain themselves is not appropriate for a policy system. For example, policies reflect procedural rules rather than embedding knowledge. Policy-based systems also do not chain rules and draw inferences as expert systems do. Expert systems are often used by specialists (e.g. a medical system for doctors) and so can benefit from detailed explanations. However, policy-based systems of the kind discussed in this paper are designed for non-experts. The explanation facility therefore needs to be simple, straightforward, and focused on the kinds of reactive rules that policies define.

1.4 Structure of The Paper

Section 2 introduces the home care system along with the goals and policies that allow it to be customised by users. Section 3 presents the approach to collecting information about the actions of the home care system and how these can be explained. Section 4 discusses how the explanation facility has been evaluated quantitatively and qualitatively by users. Section 5 summarises the work and evaluates its contributions.

2 Managing Home Care

A high-level overview is given of the ACCENT home care system. The goals and policies used to manage home care are also introduced.

2.1 The ACCENT Home Care System

The approach in this paper is called ACCENT (Advanced Component Control Enhancing Network Technologies (Turner, et al. 2012), *www.cs.stir.ac.uk/accent*). This is an approach and a set of tools for managing systems through goals and policies. ACCENT and its accompanying policy language APPEL (Adaptable and Programmable Policy Environment and Language (Turner, et al. 2014), *www.cs.stir.ac.uk/appel*) were originally developed for Internet telephony. Subsequently they have been extended into applications such as home care management and sensor networks in wind farms.

Figure 1 shows the context of a typical home care system like ACCENT. Sensors collect information about activities in the home: use of appliances and facilities, environmental and safety monitoring, movement and occupancy, etc. This information is fed into a range of services that help to deliver care in the home. The information is also recorded for local and remote analysis. Actuators support responses to sensor information: control of appliances and facilities, maintaining a comfortable environment, detecting and reporting safety-critical situations, analysing general behaviour, etc. The home is linked to a range of external services for social, health and informal care, and also to information services in general.

The ACCENT system has been developed for flexible management of a range of applications including home care. Internally, ACCENT has the high-level structure shown in figure 2. This comprises the following main elements:

Goal System: This is the primary level at which users are expected to interact with the system. Goals are high-level objectives such as staying comfortable in the home, remaining safe, or eating properly. It is expected that user goals will be identified as part of a care assessment. When external events happen, the system chooses an optimal set of policies to achieve the goals.

Policy System: This is a secondary level at which users can interact with the system. However, it is expected that only formal carers will take responsibility for policies. The policy system receives triggers (e.g. the house temperature is low, the back door is unlocked) and responds with actions dictated by the policies (e.g. turn the heating on, lock the back door).

Event Logic: Event logic can optionally be used to filter and manipulate triggers and actions. As an example of a synthetic trigger, a fall alert may not be reported unless the fall detector signals a possible fall and the user remains motionless. As an example of a synthetic action, a message to a user might be tried in several ways (as a spoken message then as a text message).

Component Infrastructure: This includes all the system components along with the general infrastructure. In the case of a home care system, it includes all the sensors, actuators and services that support care.

ACCENT is implemented as a collection of modules called bundles for the OSGi platform (Open Systems Gateway initiative, *www.osgi.org*). ACCENT supports a wide variety of sensors and actuators that are relevant to home care. The home care services include the following:

Activity: noting the user's activity around the house, e.g. movement and occupancy sensors, use of cupboards, doors and windows, and use of the toilet.

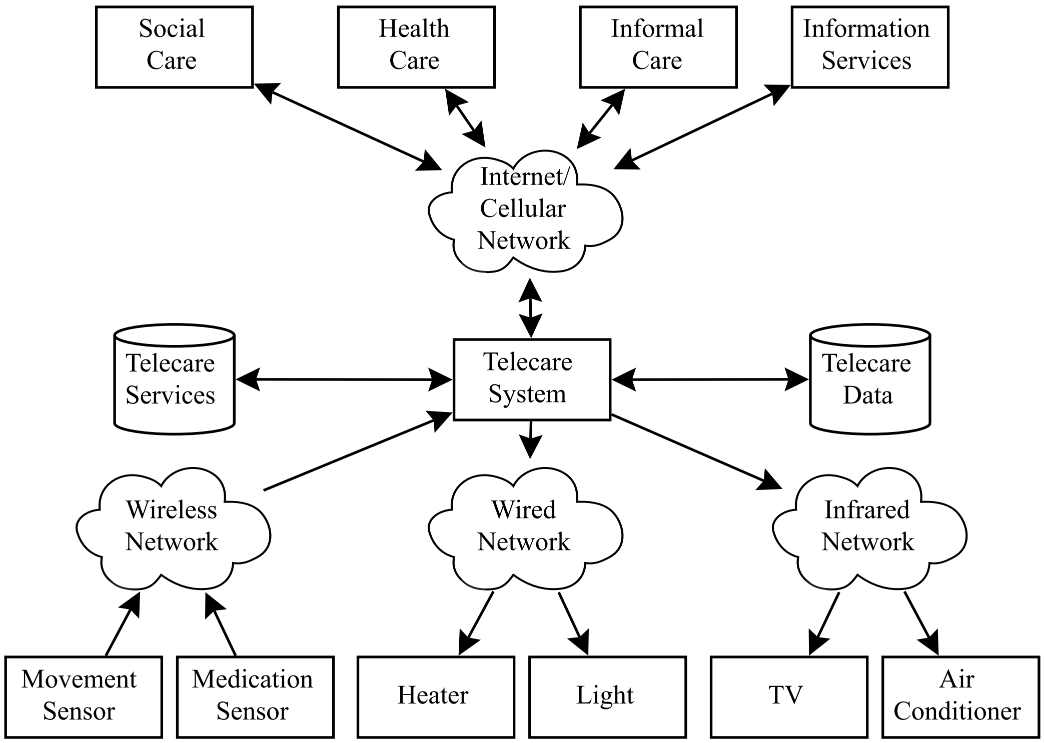

Figure 1: Example Home Care System Environment

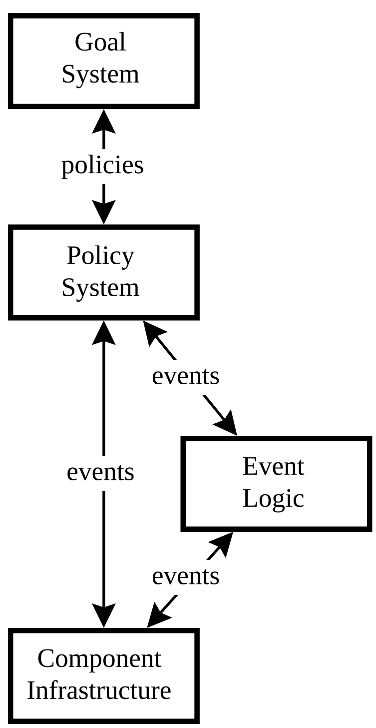

Figure 2: ACCENT System Architecture

Appliances: noting use of and controlling devices around the home, e.g. for cooking, entertainment and washing.

Communication: keeping in touch with others, e.g. by email, speech and text messaging.

Environment: control of environmental conditions in the home, e.g. air conditioning, heating, lighting, noise levels and ventilation.

Health: monitoring health-related factors, e.g. blood pressure, falls, heart rate and medication compliance.

Information: providing the user with help and reminders, e.g. about appointments, forecasts about air quality, pollen or weather, and medication guidance.

Safety: monitoring safety-related factors, e.g. detection of gas, flooding and smoke.

Security: monitoring security-related factors, e.g. door and window locking, and intruder detection.

2.2 Goals and Policies for Home Care

Home care is managed through user-defined goals and policies. Goals are user-oriented objectives for care, while policies are system-oriented ways of achieving these. These are the main way for users to manage care support. Although goals and policies can be created from scratch, a library has been developed with over a hundred predefined templates for ease of use. The user usually just needs to select a template, defining key values such as an emergency telephone number or the user's normal bedtime.

APPEL (Adaptable and Programmable Policy Environment and Language) is a comprehensive and flexible language, designed to express goals and policies in multiple domains (Turner et al. 2014). Key factors in the design of APPEL include orientation towards ordinary users, multi-lingual use, and extensibility for new applications.

Goals are defined in terms of factors that affect their achievement. Most approaches to goals are based on logic, but ACCENT treats goal achievement as an optimisation problem. This allows goals to be realised in a dynamic way depending on current circumstances (which may vary over time). It is also pragmatic in that goals are achieved as far as possible, and do not need to be completely fulfilled in some absolute sense.

As an example, suppose the user has the goal of staying comfortable. This aim is high-level and not directly executable. To give the goal meaning, the user has to specify the factors that contribute to the goal (i.e. to its measure). Each of these factors is internally associated with a weight that is automatically determined by the system so that the factors make similar contributions. Achievement of a goal is determined by a numerical measure that takes these factors into account.

As an example of factors in a goal, the user may state that being comfortable depends on the indoor temperature (with ideal value 21°C), the audio volume (noise and sound, significant only above a threshold of 70dB), and the risk of getting a chill.

A user can define multiple goals, each of which has a relative importance assigned by the user. For example, the user may decide that being secure is twice as important as maintaining social contact. The weighted combination of goal measures constitutes an overall evaluation to be optimised dynamically by the system.

Goals are achieved by special policies called prototypes. Like other kinds of policies, prototypes normally have a trigger, a condition and an action (each of which may be composite). Policies can also be associated with profiles such as 'at home' or 'on holiday'. Unlike other kinds of policies, prototypes indicate how they affect goals through their effect on the factors in goal measures.

As an example, a prototype might be defined to ensure that the house does not become overheated. If the indoor temperature is reported as hot (a trigger) and the house is occupied (a condition), the air conditioning can be set to high (an action). This might have the effects of reducing the temperature by 4°C and increasing energy consumption by 4kWh. These could be factors in achieving goals such as being comfortable or saving energy.

A regular policy is similar to a prototype but does not identify an effect on goals. This is used for policies that should always apply irrespective of the current goals. As an example, if an older person needs to go to the toilet at night then there is a risk that they will fall in the darkness. A policy can be defined to monitor a bed occupancy sensor. If this reports that the user has risen (a trigger) at night (a condition), the toilet light would be switched on (an action). When the bed becomes occupied again, the toilet light would be switched off.

When triggers occur, they cause regular policies and goal-related (prototype) policies to be activated. Goal-related policies are selected to maximise the overall evaluation function: a numerical measure of how well all the goals are being achieved. The policies that best contribute to the goals are then combined with regular policies. This allows the system to react appropriately and dynamically to changing circumstances.

As multiple policies can be triggered, conflicts may arise among their actions. The policies of just one user may conflict, perhaps due to contradictory goals like saving energy but staying comfortable. More typically, conflicts arise due to policies defined by different people (e.g. the end user, a family member or an occupational therapist). APPEL supports special resolution policies that deal with conflicts like these. Resolutions automatically detect conflicts and decide how to resolve them.

Users interact with the policy system via a number of alternative interfaces. For example, there are several wizards that ease the task of defining and editing goals and policies. The web wizard is the most comprehensive: it is multi-lingual and can be used remotely (e.g. by a care worker). The wizard supports a near-natural (though somewhat stilted) language interface that is multilingual (being currently localised in English, French and German).

Kenneth J. Turner. Explaining the Operation of A Home Care System
J. Assistive Technologies, 9(1):21-37, April 2015, Emerald Group Publishing

As an example of this wizard, Figure 3 shows the goals currently defined by a user. Goals can be edited by clicking on their labels, can be disabled, and can be deleted. The relative importance of each goal can be set by dragging a slider.

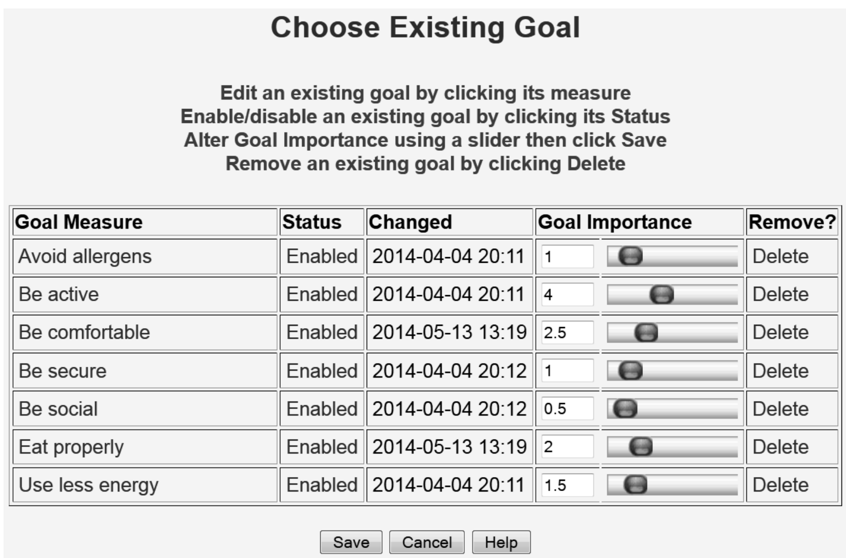

Figure 3: Editing A List of Goals

As a further example of the web-based wizard, Figure 4 shows a prototype policy that heats the house as required. This is owned by the system administrator for Computing Science Stirling, and applies to all users in this domain. The policy has the effect of increasing the interior temperature by 5°C and increasing energy consumption by 3kWh. When the interior temperature is read, if this is cold then the toilet heater is set to high.

3 Explaining The Home Care System

It is discussed how the policy system keeps a history of policy executions. This is part of the support for queries about past and future behaviour of the system.

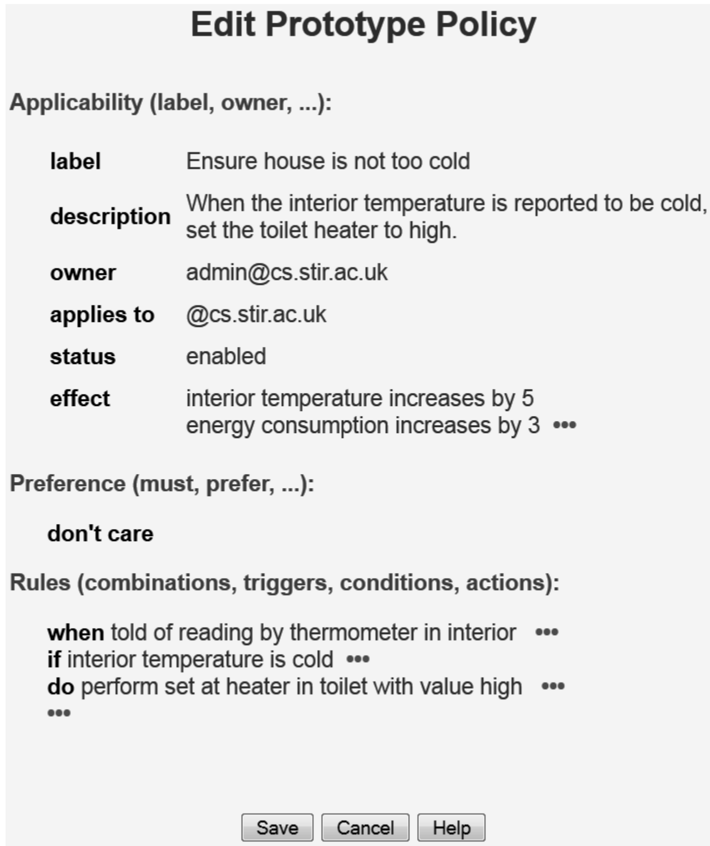

Figure 4: Editing A Prototype Policy

3.1 Policy System History

For the work reported in this paper, a key addition to the policy system has been keeping track of policy processing. The stages in policy execution are shown in figure 5.

The processing is started by arrival of a trigger (e.g. a signal from a cardiac monitor or a fall detector). The corresponding user's policies are then retrieved and checked against this trigger. Triggered policies have their conditions checked (e.g. for heart rate or the time of day). The resulting applicable policies are then separated into regular ones and goal-related ones. A selection of the latter is made to optimise achievement of all the goals. The optimised policies are then combined with the regular ones. The actions of these policies are extracted and checked for conflicts. This results in a final set of actions that are optimal and conflict-free. The home care system then executes these actions (e.g. verbally advising the user to rest or sending an alert message to a carer).

The policy system keeps a detailed history of such activations. There is a user-defined limit on how long these are preserved, since it would not be meaningful to query old actions (say, over a day ago).

3.2 Policy Explanation Interface

For the work reported in this paper, a policy explanation interface has been implemented as an extension to the web-based policy wizard. For convenient formulation of queries, this reuses the wizard's ability to define policies, triggers and actions. As shown in figure 6, three kinds of policy queries are supported. The user fills in the details for one type of query and clicks Check to see an explanation.

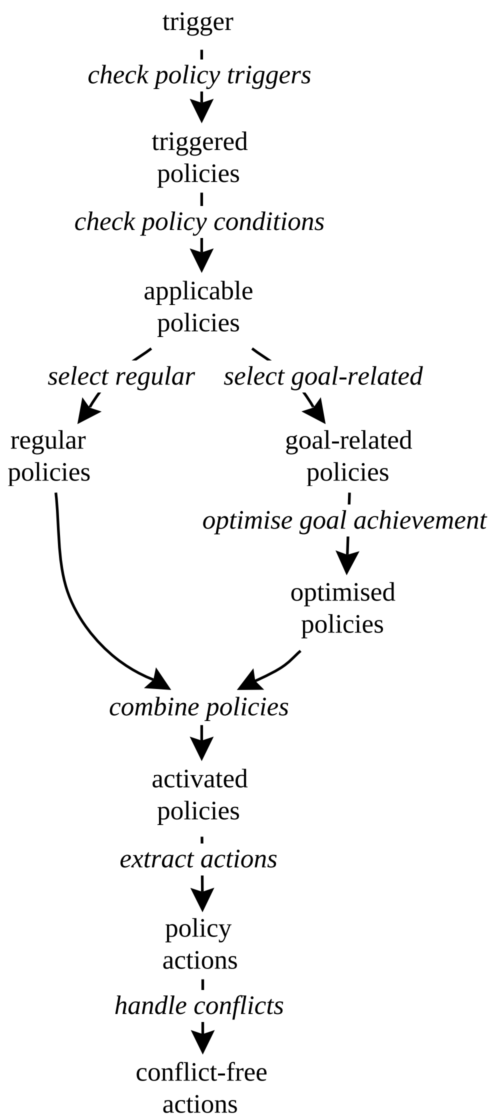

Figure 5: Stages in Policy Execution

3.2.1 Policy History

As time progresses, a number of policies may be accumulated for the home. These may be defined by a variety of people: the end user, a family member, an informal carer or a care professional. Some of these policies may be temporary (e.g. while the user is on holiday) and so may become obsolete in time. It is therefore natural to ask what policies have been defined and when they have been used. It is also interesting to know if some policies are not being used and why this should be.

Figure 7 shows the query interface for policy execution. A regular policy or prototype policy is selected from the drop-down list and then checked. In the figure, the policy 'Avoid cold house' has been queried. As will be seen, this has been triggered three times but was not executed on one occasion because its conditions were not met. If this non-execution is a surprise, the user can investigate why the conditions were met. For example, the policy may not have been applicable for that day.

For a prototype policy, the analysis also reports whether it was optimal or not in terms of the goals. This can be useful in tuning goal weights. Suppose the prototype 'Ensure house is not too hot' is queried. This contributes positively to the goal 'Be comfortable' because it avoids overheating. However, it contributes

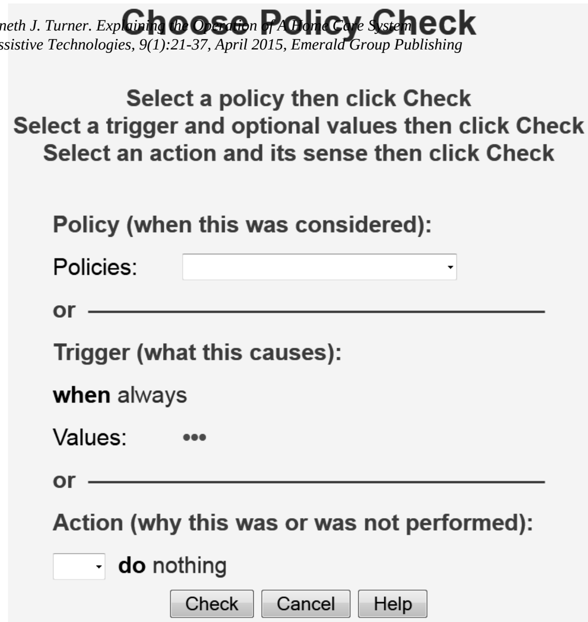

Figure 6: Policy Checking Interface

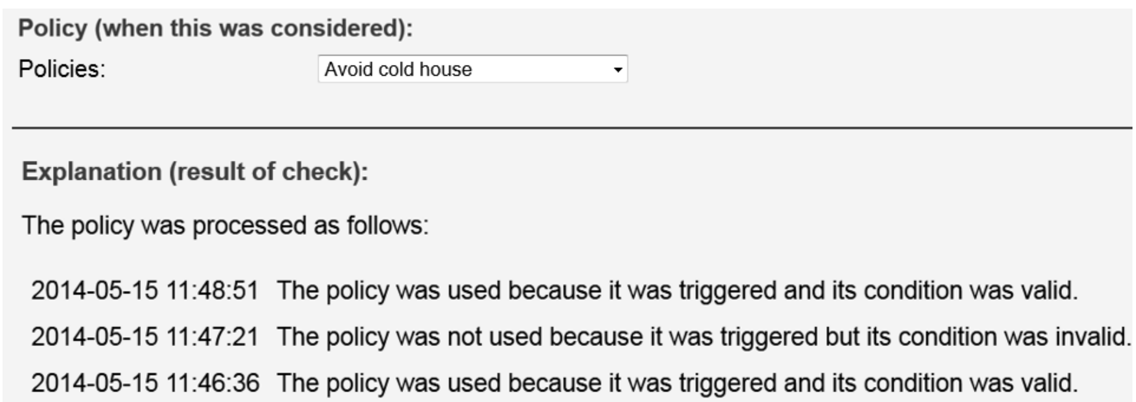

Figure 7: Explanation of Policy Execution

negatively to the goal 'Use less energy' because it uses the air conditioning to cool the house.

When this prototype is queried, the report might say that it was not executed because it was not optimal for the goals. For example, this could indicate that the importance of goal 'Be comfortable' is too low relative to that of goal 'Use less energy', so the prototype was not selected. The user might therefore decide to increase the importance of 'Be comfortable' so that such a prototype is selected.

3.2.2 Trigger Outcome

After defining a policy, the user may be interested to know what will happen if it is triggered. In general, the user will have questions about what the system will do: what will happen when I come home? who will be called if I fall? what will the result be of getting up at night? A further query capability therefore allows a trigger to be defined and then simulated. The policy system processes the trigger as if it were real, and reports the actions that would result.

Figure 8 shows an example where the trigger has been defined as the porch door opening at 1.15 AM. The trigger would result in three actions from the regular policy 'Discourage night wandering'. This is

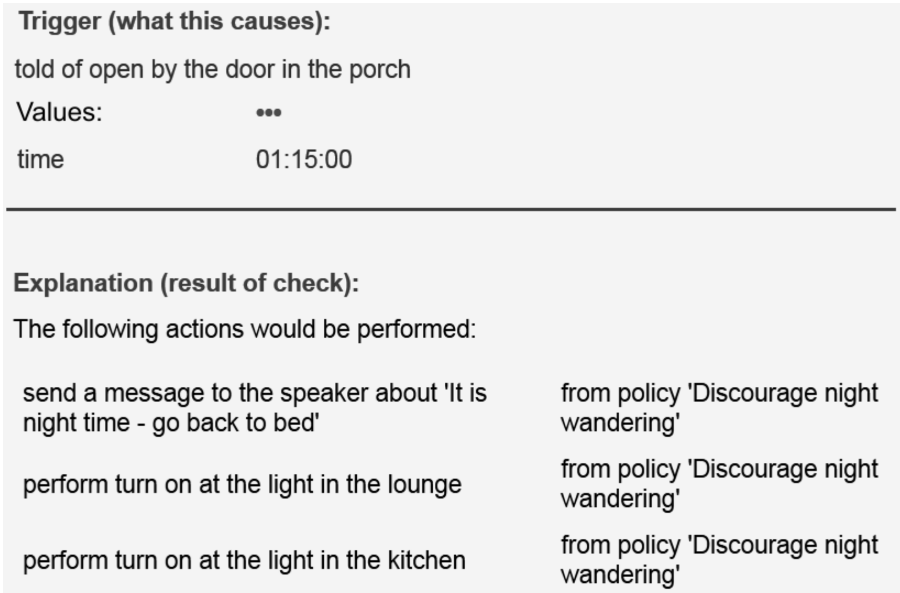

Figure 8: Explanation of Trigger Results

designed to handle the common situation of an older person getting up in the middle of the night and thinking it is time to go out. The explanation shows that the user would be given a spoken reminder to return to bed. The lounge and kitchen lights would also be turned on to encourage the user to go back into the house.

This example shows the optional use of values along with a trigger. Some policies depend on environment values such as the exterior temperature or previously recorded information. One or more environment values such as 'time' here can therefore be defined for policy simulation.

One or more actions (perhaps from different policies) might result from a trigger. If it is found that a trigger produces no actions, this may need to be investigated. Perhaps no policy has the specified trigger, or policies were triggered but their conditions did not hold, or there were applicable policies but they were non-optimal. If a particular action was expected, it might turn out that this was omitted because it conflicted with other actions. For example, turning on the lounge light might be prohibited by a policy that keeps downstairs lighting off at night (to save energy).

3.2.3 Action Execution

As the home care system operates invisibly in the background, the user might wonder at times why a particular action occurred (or did not occur). The user may therefore have questions about the system: why is the heating on? why was I asked to close the windows? why was a TV programme recorded?

Figure 9 shows an example where the user has asked about the heating being set to warm. The policy 'Avoid cold house' is reported as the source of this action. However, the explanation goes beyond a literal answer as to when and why some action occurred. It may also be reported that a different action was performed more recently for the same device. In the figure, policy 'Avoid cold house' was executed later in different circumstances (the house was colder), resulting in the heating being set to hot. This avoids the user being misled that the action has been performed as expected. It may, of course, be reported that the expected action has not been performed at all. The user can then investigate why this is the case.

More interestingly, it is also possible to ask why actions were not performed: why is the heating not on? why was I not asked to close the windows? why was a TV programme not recorded? Such a query is formulated by selecting 'not' in front of the action. The explanation might report that the expected action was performed, but that it was overridden by a different action for the same device. It might also be reported that the action was indeed performed recently, or that no action for this device has occurred. Again, the

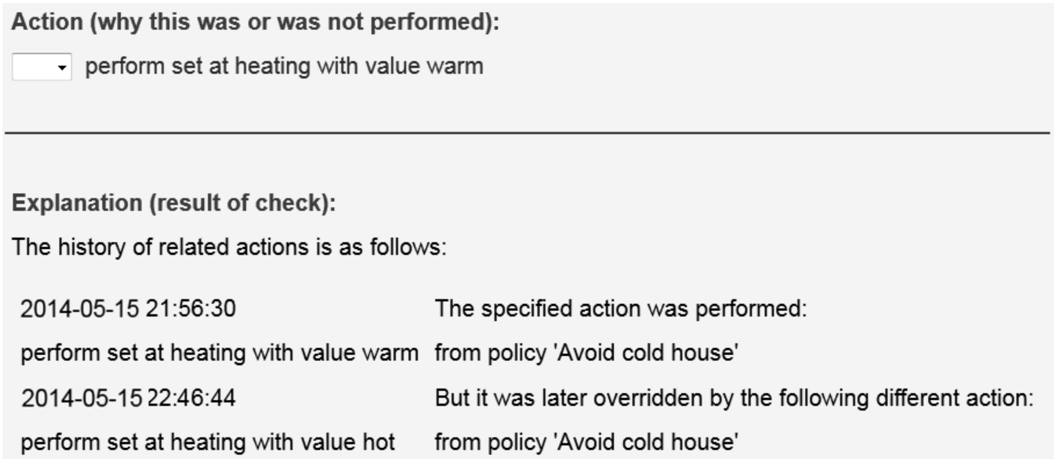

Figure 9: Explanation of Action Execution

user can investigate why something did not happen.

4 Evaluation of The Explanation Facility

The usability of the explanation facility has been assessed through a mixed empirical evaluation using quantitative and qualitative analysis. At this stage, only a preliminary evaluation has been conducted in order to guide future development.

The aim was to check the following hypothesis: someone with a care background and basic computing knowledge, with 45 minutes of training on the approach, can use the explanation facility effectively. Effective use was taken to mean answering nine questions about system operation with 80% accuracy in at most 25 minutes. This hypothesis reflected the author's aspiration that the approach be easy to learn and easy to use.

The author recruited five carers without previous experience of the approach to be evaluated. All were female (as is common in caring), with average age 41 (range 21 to 63). Their roles were telecare manager, care home manager, two care home assistants, and a care developer. Only the last of these had significant knowledge of computing. The participants were given a one-on-one introduction to the home care system in general and to the explanation facility in particular. It was expected that this familiarisation phase would take about 45 minutes.

Participants were then asked to work unaided through nine questions that required use of the explanation facility. It was expected that this exercise phase would take about 25 minutes. The home care system was initially set into a definite state (known to the author but not the participants) where various triggers and actions had already been processed. Participants were then asked to answer various questions about what the system had done and why. This required translating the natural language questions into settings for the explanation facility, and then interpreting the responses from this to answer the questions. The following examples illustrate the kinds of questions that were posed.

- Why has the policy 'Cool house naturally' not been used?
- Has the policy 'Ensure house is not too cold' ever contributed to the goals?
- What will happen when the interior temperature is reported to be 15°C?
- What will happen if the front door is opened at 1 AM?
- Why is the bedside light on?
- Why is the standard light in the lounge not off?

The participants were then asked to rate five statements about the approach on a five-point Likert scale. They were also given the opportunity to provide verbal comments about the approach.

The times needed for the introduction and exercise phases were noted. The participants spent an average of 31 minutes (range 24 to 39) on the familiarisation phase. This compares favourably with the author's expectation of 45 minutes. The participants spent an average of 14 minutes (range 7 to 25) on the exercise phase. This compares favourably with the author's expectation of 25 minutes. Predictably the care developer (who had software experience) achieved the shortest times.

The author also scored the accuracy with which the participants posed questions to the explanation facility and correctly interpreted the results. Their accuracy was an average of 95% (range 94% to 97%). Interestingly, all participants (irrespective of their computing knowledge) performed very similarly.

The participants were asked to rate five statements about the approach on a scale from 1 (strongly disagree) to 5 (strongly agree). These statements were designed to elicit qualitative information about the usability and comprehensibility of the explanation facility.

Statement 1: *I found it straightforward to answer questions about policies*: average score 4.6 (range 4 to 5).

Statement 2: *I found it straightforward to answer questions about what would happen on a trigger*: average score 4.4 (range 4 to 5).

Statement 3: *I found it straightforward to answer questions about what actions have happened*: average score 4.6 (range 4 to 5).

Statement 4: *I think the approach would help to understand the rules for home care*: average score 4.4 (range 4 to 5).

Statement 5: *If I had the home care system, I would use its ability to check policies*: average score 4.6 (range 4 to 5).

In their verbal comments, the participants also provided valuable feedback on how the approach could be improved. Some weaknesses in the current explanation facility emerged from this evaluation. A future version of the explanation facility will address these suggestions:

- The system uses technical terms such as 'goal' and 'policy'. One participant noted that these have specific meanings for care providers and suggested that more appropriate terminology be used.
- Negative actions are formulated by selecting 'not' in front of the action. Two participants suggested that it would be clearer to offer an explicit choice of positive and negative actions.
- One participant was unclear about what 'value' means. In the current interface, this refers to the parameter of a trigger (such as a temperature of 15°C) and also to an environment value associated with a trigger (such as a time of 1 AM).
- The current interface is intentionally open-ended about choices for environment values since these have many purposes. As a result, one participant was unsure how to name the value for the current time (e.g. it might have been 'hour' or 'time').
- One participant observed that 'set' has two different meanings: for setting an appliance to some state (e.g. setting the heating level) and for setting a variable to record some information (e.g. that the house is occupied).

Overall, the participants performed well despite the limited time spent on familiarisation. The evaluation required the participants to learn new concepts and a new interface in a comparatively short period. If the explanation facility were to be deployed in actual practice, extended training would be offered. Although the limited number of participants does not allow statistically meaningful conclusions, the results of this initial evaluation are encouraging and favour the author's hypothesis about ease of learning and use.

5 Conclusion

5.1 Summary

It has been argued that home care technologies offer significant benefits in supporting older people to live independently. Goals and policies have been described as a means of allowing users to state how a home system should automatically support their care needs. These rules are interpreted by a home care system that is integrated with a wide variety of telecare and telehealth devices. An explanation facility has been described that allows the operation of the home care system to be queried. Specifically, this deals with 'when' questions about policy usage, 'what if' questions about the consequences of triggers, and 'why' questions about automated system actions.

5.2 Evaluation

The home care system is designed to be unobtrusive. For example, it will run on a headless computer that can be placed out of sight and it will also run on a Raspberry Pi for a compact installation. The system can be used with wireless devices or with devices connected to existing mains wiring, so the disruption of an installation can be minimised. System rules are normally defined by a carer using the library of goal and policy templates. Defined rules can easily be altered within the home or using secure remote access. Care data and alerts can also be accessed within the home or securely from another location.

Rules for automated support would normally be identified following a care assessment, and so would usually be defined by a social or health care worker. For example it may be determined that the end user needs reminders about appointments and meal times, that a fall detector should be used, and that the user's fluid consumption needs to be monitored. There is also scope for informal carers (e.g. a family member) to define rules as well and to receive alerts. If conflicts arise among rules, the system is able to give priority to (say) the rules defined by a care professional.

The ACCENT home care system has been separately evaluated for usability of various aspects: robustness, event logic, policies, goals, and now explanation.

An initial evaluation was performed of system robustness, usability of event logic, and usability of policies (Turner 2011). For robustness testing, the system has been deployed for the past three years in the homes of two volunteer users (ordinary householders rather than vulnerable people). The new explanation facility has been operational for nine months. Apart from some weaknesses in third-party driver code (now corrected), the system has been found to behave reliably.

In (Turner 2011) the use of event logic was evaluated for graphical description of care services that integrate multiple sensors and actuators. It has been shown that this could be quickly learned and used effectively. The use of policies was also evaluated, showing that these were understandable by non-technical users. In a later study (Maternaghan 2012), it was found that a wide variety of users appreciate a policy-based approach and the power that it confers. This demonstrates that ordinary users are able to relate to policies and to formulate them successfully.

The usability of home care goals has been evaluated with a group of care managers as the most likely kind of user to define these. A short scenario was provided to describe the situation of a hypothetical older couple. Participants were then asked to formulate home care goals for this couple, and to define policies that could be used to realise these. All participants were able to come up with plausible goals for home care. Perhaps more surprisingly (because it is a more technically challenging task), all participants successfully thought up policies that could be used to achieve the goals.

For the work reported in this paper, a new usability evaluation has shown that the explanation facility has promise. Care workers are able to learn how to use this relatively quickly, and can answer questions effectively about system operation. The work is believed to offer a new way of interrogating the operation of a policy-based system. Although there are affinities with the work on explanation in expert systems, the nature of policies requires a significantly different solution.

Of course the system described in this paper is only a research prototype. However, it is hoped to influence the design of future commercial systems to incorporate the ideas that have been elaborated in this work.

5.3 Future Work

Section 4 identified some improvements that need to be made in the explanation system, mainly in using appropriate terminology so that it is more understandable. It is believed that the technical capabilities for explanation are adequate to present most aspects of how the system behaves. However more explanations could be added, such as what services and devices are currently available in the system.

At present the system explains its actions using the web wizard. For those with a sensory impairment, an obvious extension is to use the existing speech synthesis capability for spoken explanations. However, this should be coupled with speech recognition to allow the user to make verbal requests for information about the system. Fortunately the range of queries is restricted so the limited speech recognition currently supported (using the Google speech API) should be sufficient.

In time it is planned that the home care system be evaluated in the homes of 'real' end users. Discussions are under way with a Local Authority to conduct such a trial.

Acknowledgements

The author is grateful to the participants who helped with the usability evaluation.

References

J. E. Bardram (2004). 'The Java Context Awareness Framework – A Service Infrastructure and Programming Framework for Context-Aware Applications'. In H. W. Gellersen, R. Want, & A. Schmidt (eds.), *Proc. 3rd Int. Conf. on Pervasive Computing*, no. 3468 in Lecture Notes in Computer Science, pp. 98–115. Springer, Berlin, Germany.

N. Damianou, et al. (2001). 'The Ponder Policy Specification Language'. In *Policy Workshop 2001*, no. 1995 in Lecture Notes in Computer Science. Springer, Berlin, Germany.

K. Doughty, et al. (1996). 'Three Generations of Telecare of The Elderly'. *Telemedecine and Telecare* **2**(1):71–80.

T. Dursun & B. Örencik (2003). 'Police: A Novel Policy Framework'. In *Proc. ISCIS 2003*, no. 2869 in Lecture Notes in Computer Science, pp. 819–827. Springer, Berlin, Germany.

S. Helal, et al. (2012). 'Towards An Ecosystem for developing and programming Assistive Environments'. *Proc. IEEE* **100**(8):2489–2504.

C. Leong, et al. (2009). 'A Rule-Based Framework for Heterogeneous Subsystems Management in Smart Home Environment'. *IEEE Transactions on Consumer Electronics* **55**(3):1208–1213.

C. Maternaghan (2012). 'Can People Program Their Home?'. Tech. Rep. CSM-191, Computing Science and Mathematics, University of Stirling, UK.

C. Maternaghan & K. J. Turner (2011). 'Pervasive Computing for Home Automation and Telecare'. In S. I. A. Shah, M. Ilyas, & H. T. Mouftah (eds.), *Pervasive Communications Handbook*, pp. 17.1–17.25. CRC Press, Boca Raton, Florida, USA.

J. D. Moore & C. L. Paris (2007). 'Requirements for An Expert System Explanation Facility'. *Computational Intelligence* **7**(4):367–370.

J. D. Moore & W. R. Swartout (1988). 'Explanation in Expert Systems: A Survey'. Tech. Rep. ISI/RR-88-228, Information Sciences Institute, University of Southern California, Marina del Rey, California, USA.

M. Munstermann, et al. (2012). 'A Novel Human Autonomy Assessment System'. *Sensors* **12**(6):7828–7854.

M. Román, et al. (2001). 'Gaia: A Middleware Infrastructure for Active Spaces'. *Pervasive Computing* **1**(4):74–83.

Scottish Executive (2001). *Guidance on Single Shared Assessment of Community Care Needs (Circular CCD 8/2001)*. Scottish Government, Edinburgh.

Select Committee on Economic Affairs (2003). *Aspects of The Economics of An Ageing Population*. The Stationery Office Limited, London.

L. S. Shafti, et al. (2013). 'Inferring ECA-Based Rules for Ambient Intelligence using Evolutionary Feature Interaction'. *Ambient Intelligence and Smart Environments* **5**(6):563–587.

W. R. Swartout (1983). 'XPLAIN: A System for creating and explaining Expert Consulting Systems'. *Artificial Intelligence* **21**(3):285–325.

K. J. Turner (2011). 'Flexible Management of Smart Homes'. *Ambient Intelligence and Smart Environments* **3**(2):83–110.

K. J. Turner, et al. (2012). 'Goals and Policies for Home Care'. In K. J. Turner (ed.), *Advances in Home Care Technologies: Results of The* MATCH *Project*, pp. 3–49. IOS Press, Amsterdam, Netherlands.

K. J. Turner & C. Maternaghan (2012). 'Home Care Systems'. In K. J. Turner (ed.), *Advances in Home Care Technologies: Results of The* MATCH *Project*, pp. 11–29. IOS Press, Amsterdam, Netherlands.

K. J. Turner, et al. (2014). 'APPEL: Adaptable and Programmable Policy Environment and Language'. Tech. Rep. CSM-161, Computing Science and Mathematics, University of Stirling, UK.

K. J. Turner, et al. (2006). 'Policy Support for Call Control'. *Computer Standards and Interfaces* **28**(6):635–649.