

# Semantic-Based Decision Support for Remote Care of Dementia Patients

Taha Osman, Ahmad Lotfi, Caroline Langensiepen,  
Saisakul Chernbumroong  
School of Science and Technology  
Nottingham Trent University  
Nottingham, United Kingdom  
{taha.osman, ahmad.lotfi, caroline.langensiepen,  
saisakul.chernbumroong}@ntu.ac.uk

Mahmoud Saeed  
John Black Day Hospital  
Birmingham, United Kingdom  
mahmoud.saeed@bsmhft.nhs.uk

**Abstract**—This paper investigates the challenges in developing a semantic-based Dementia Care Decision Support System based on the non-intrusive monitoring of the patient’s behaviour. Semantic-based approaches are well suited for modelling context-aware scenarios similar to Dementia care systems, where the patient’s dynamic behaviour observations (occupants movement, equipment use) need to be analysed against the semantic knowledge about the patient’s condition (illness history, medical advice, known symptoms) in an integrated knowledgebase. However, our research findings establish that the ability of semantic technologies to reason upon the complex interrelated events emanating from the behaviour monitoring sensors to infer knowledge assisting medical advice represents a major challenge. We attempt to address this problem by introducing a new approach that relies on propositional calculus modelling to segregate complex events that are amenable for semantic reasoning from events that require pre-processing outside the semantic engine before they can be reasoned upon. The event pre-processing activity also controls the timing of triggering the reasoning process in order to further improve the efficiency of the inference process. Using regression analysis, we evaluate the response-time as the number of monitored patients increases and conclude that the incurred overhead on the response time of the prototype decision support systems remains tolerable.

**Keywords**—*Semantic Web; Dementia; decision support; remote healthcare; knowledge Inference; context awareness; complex event processing*

## I. INTRODUCTION

Cognitive decline and the related diseases such as Dementia and Alzheimer’s disease (AD) are becoming increasingly common in the geriatric population. Worldwide, 34 million people suffer from Dementia and in UK alone this figure is nearly 700,000 [1]. The cost of Dementia to the UK economy is approximately 17 billion GBP each year [2] and is expected to rise significantly with the aging population trend; the number of people over the age of 60 is expected to rise from 33% of those aged between 15 and 60 in 2001 to 55% in early 2030s[3].

In the present scenario, care is given to Dementia sufferers by NHS in the form of personal attendants such as nurses and

social workers. Reducing the amount of personal care devoted to early Dementia sufferers by means of remotely monitoring their condition not only reduces the pressure on NHS resources, but also promotes good quality independent living of the Dementia sufferer.

Remote healthcare for Dementia suffers has attracted a lot of investment. A large body of commercial and research efforts in the field concentrated mainly on memory loss, which is a main characteristic of Dementia. Such systems offer guidance to the patients in their day-to-day activities such as reminding the patients to take medicine [4][5]. Our research subscribes to the more challenging efforts that deploy sensory devices to monitor the activities of daily living of early Dementia suffers, and then analyse the sensory output to identify deviations from normal behavioural patterns that can indicate a deterioration in the Dementia condition such as restlessness and wandering, and subsequently advise intervention from the caregivers or the health clinicians. In this paper we argue that knowledge-based systems are well-suited for analysing the recorded patients’ behaviour, and offer advice to a corresponding Dementia Care Decision Support Systems (DCDSS) about probable deterioration in the condition. In the long term, the knowledgebase can provide new insight into the data resulting from correlating information from different/multiple sources, such as inferring the impact of a new drug by reasoning over recorded patient’s behaviour.

The research documented in this paper investigates the feasibility, from a technical viewpoint, of utilising Semantic Web technologies to develop a knowledge-based approach to Dementia care decision support. Based on earlier effort published in [6] that studied Semantic Web reasoning over simple (atomic) events for context-aware DCDSS, the research reported in this paper focuses on the interactivity dynamics of the semantic reasoning engine with the complex events streamed from the sensory equipment, in order to support knowledge inference that fulfils the requirements for remote healthcare decision support.

We facilitate the utilisation of Semantic Web Rule Language SWRL [7] in complex event processing by adopting a new approach that relies on propositional calculus modelling to segregate complex events that are amenable for semantic

reasoning from events that require pre-processing outside the semantic engine before they can be reasoned upon, and perform experimental analysis that deploy regression analysis to evaluate the performance of the semantic reasoning process.

The rest of this paper is organised as follows: section 2 presents the motivation behind undertaking this research. Section 3 details our semantic modelling and knowledge mapping approach. The challenges of complex event processing in Semantic Web systems are discussed in section 4, leading to presentation of the framework for semantic decision support framework in section 5. The following section details the presentation and evaluation of the experimental results, and section 7 concludes the paper and presents our plans for further work.

## II. MOTIVATION

Ongoing investigation in our research group is involved in developing an Intelligent Inhabited Environment (IIE) infrastructure with the purpose of monitoring the activities of daily living of elderly occupants and assisting them in leading a good quality independent living [8]. The architecture of the IIE is illustrated in Fig 1, where the data collected from the sensor network are wirelessly communicated with a base station and eventually stored in a central database. The IIE relies solely on non-intrusive sensors such as door and infrared motion sensors; research has proven that the use of intrusive technology such as surveillance cameras for patient monitoring is generally not accepted [9]. The sensors produce long series of binary multidimensional data that are difficult to analyse and manipulate manually, so a combination of Principal Component Analysis and Fuzzy Rule-based Systems are applied to process the data and identify and also predict deviations from normal behaviour pattern [10], which are relayed to the health clinicians and the patient carer.

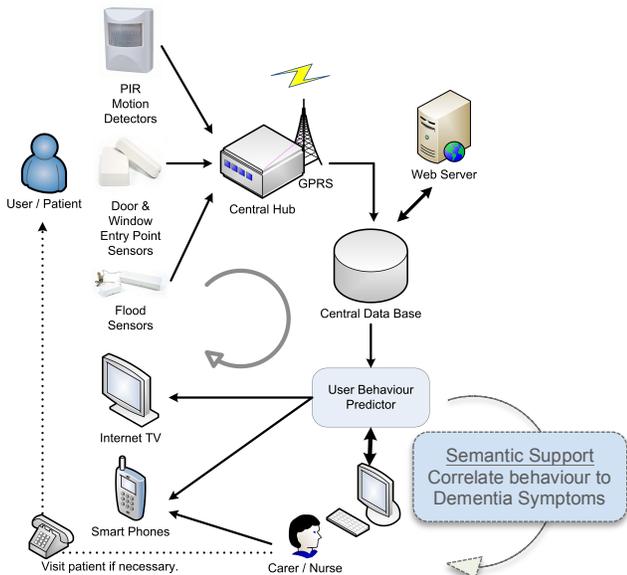


Fig. 1. An overview of the IIE architecture with the semantic decision support extension

The research documented in this paper attempts to correlate the identified behavioural patterns to the symptoms of

Dementia with the long term objective of developing a semantically-enabled Dementia care decision support system (DCDSS) that can aid health clinicians in the remote assessment of the patients' condition, and thus reduce the demand for on-site care and support longer independent living for the Dementia sufferers. To achieve this, the decision support software needs to not only possess a human-like understanding of the complex concepts of the domains involved in a remote care system such as sensory movement information, human behaviour, disease diagnoses, and clinical guidelines, but also be able to intelligently interrelate these concepts to assist medical advice. Semantic Web technologies are ideally suited to model such complex systems; In contrast to static terminology structures used for knowledge reference, semantic ontologies allow to describe at the taxonomy level the concepts and sophisticated relationships between the concepts (subsumption, cardinality, jointness, etc.), thus allowing for knowledge inference and reasoning [11]. Semantic knowledgebases are also inherently extensible [12] compared to traditional RDBS, which is again very useful for the ever-evolving medical domains.

## III. SEMANTIC APPROACH TO MODELLING CONTEXT-AWARE DEMENTIA CARE DECISION SUPPORT SYSTEM

Our approach to modelling the Dementia Care Decision Support System (DCDSS) is based on the premise that dynamic changes in the patients' situational conditions (Context) such as room occupancy and movement within the living environment has to be analysed against the semantic knowledge about the condition's symptoms and the patient's related profile (medical history, age, etc.) in order to interrelate them to Dementia conditions such as restlessness.

Hence we can accurately describe our DCDSS as context-aware and knowledge-based. The use case examined in this study captures change in the situational conditions such as shorter room occupancy and more frequent movement between the rooms, which is analysed against the knowledge of the patient history of restlessness and inferred as deterioration in the patient's Dementia condition. Moreover, as more patients are diagnosed with Dementia, recording the patients' data in an intelligent semantic knowledgebase becomes increasingly important. In contrast to the temporal advice of traditional decision support systems, such knowledgebase can provide for inference over data that is collected over longer periods and covering a larger population of Dementia sufferers. Such inference can, for instance, provide new insight into the long-term impact of certain medicine on the well being of the patient.

As demonstrated in the use cases discussed in this section, the value proposition of a knowledge-based approach is in applying intelligent reasoning to a domain in order to solve a problem that would otherwise require considerable effort and expertise. Building the knowledgebase requires mapping domain knowledge to the requirements of the problem, and then using the mapped knowledge to acquire additional information about the problem domain and infer new facts that aid the solution [13]. Hence, we initially make the following assumptions about the system requirements:

- The primary end-users of the system are health clinicians: doctors, nurses, and carers, who monitor the condition of the patient remotely.
- The system should differentiate between alarming and critical anomalies in patient behaviour with critical anomalies being informed with a higher priority and in a way that demands quicker response.
- The Dementia patient is the sole occupier of the monitored residence. This significantly simplifies the movement and occupancy analysis.
- The Dementia residence consists of four rooms, a lounge, bedroom, kitchen, and bathroom with one main exit door.

Guided by the above requirements we gathered the domain knowledge, aided by external sources such as the Alzheimer’s Association’s ‘2010 Alzheimer’s disease facts and figures [14], and also by eliciting knowledge from a specialist psychiatrist who had the practical experience in managing the care for Dementia patients and can also represent the prime beneficiary of the envisaged DCDSS. The gathered knowledge was conceptualised using a concept map as illustrated in Fig 2, which was later translated into formal semantic ontology representation.

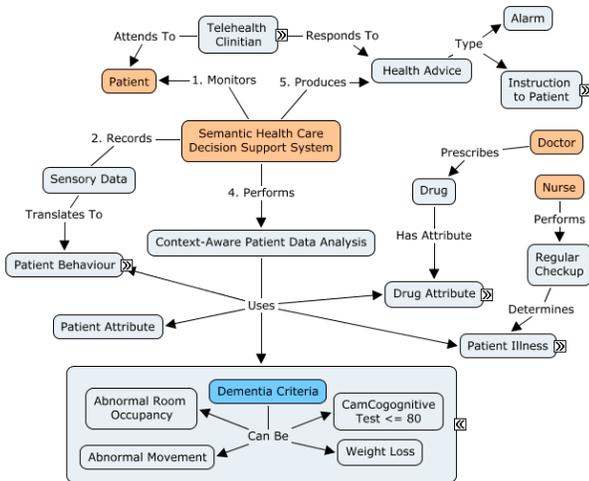


Fig. 2. Concept Map for the Semantic-Driven Decision Support System

The “Semantic Health Care DSS” uses various sensors to capture ‘events’ related to the monitored patient’s behaviour (such as room occupancy and equipment use) to determine the context of the patient. The system then performs context-aware patient data analysis on the captured events and produces health advice. Therefore, it is clear that event processing is a critical process in the decision support system.

In contrast to works that provide for reasoning over atomic events such as in [22], in this work we investigate reasoning over a stream of complex interrelated events, which represents a major challenge for the Semantic Web technology [15] that is primarily geared to model static knowledge. The findings of our earlier research published in [6] established that Semantic rules such as SWRL can be seamlessly integrated into the semantic ontologies but do not support the event workflow

capabilities of external Event Processing languages such as EP-SPARQL [16], which on the other hand add to the complexity of building the decision support system. The research approach reported in this work advocates the deployment of an external proprietary events’ pre-processor that can efficiently handle the temporal attributes of events outside the semantic knowledgebase.

#### IV. INVESTIGATION OF COMPLEX EVENT PROCESSING IN SEMANTIC WEB SYSTEMS

In this section, we introduce a new approach for handling complex event processing within the Semantic Web framework. We intend to deploy propositional calculus modelling to segregate complex events that are amenable for semantic reasoning from events that require pre-processing outside the semantic engine before they can be reasoned upon.

##### A. Event metadata

Complex events can be generated from atomic events by combining them in different ways. When combining multiple events, various event metadata are processed such as the event type, occurrence time and location, which form part of the event context. This work is concerned with the following metadata of events:

##### Event attributes

- E1: Timestamp of an event
- E2: Location of an event (the context)

##### Relationships between events:

- E3: Time interval between any two events
- E4: Change in context between any two events

We also introduce another special type of event, which is the passage of time. This event, termed ‘E5’, is generated every ‘n’ seconds in order to aid in determining the time intervals of interest.

Finally, we model the response time to the occurrence of the event as follows:

- T1: Events that need an immediate response as soon as they are detected
- T2: Events that require response after a certain time interval (E5) from the moment they are detected

To summarise, any event has two dimensions, the metadata of the event, and the relationship of the event to other events including the time passage event.

##### B. Use-cases for patients with Dementia (PWD)

This section associates specific event types related to the PWD use-case with the event metadata identified in the previous section. There are events that will trigger response from the decision support system with out the need to evaluate the E1-E5 event metadata; for instance, an ‘accidental fall’ would trigger response from the system irrespective of the time and location at which the event occurred.

Below we consider events (such as opening main door at night, weight loss, etc.) that require the evaluation of a single or combination of the modelled event metadata (E1-E5).

E1 - These are events that become significant because of the time at which they occur; for instance:

- Opening the main door (event type) at midnight (E1: event timestamp). PWD tend to wander outside at night, which is widely accepted as an abnormal and a potentially dangerous situation that needs to be avoided.

E2 - These are events where the location of the event is of significance that become significant because of location at which they occur, similar to:

- Prolonged stay (event type) inside bathroom (E2)

E3, E4 - These are events that take a combination of the change in time intervals and the change in contexts between events, such as:

- Restlessness (event type) is calculated by determining quick (E3) changes in contexts (E4)
- Restlessness (complex event type) followed by main door open (event type). This is significant because PWDs tend to become aggressive and attack passers-by
- Frequent bathroom usage (complex event type using E3 and E4) with or without high diabetes level (event type)

E1, E5 - This combination is used to determine if, from the set of events (event types) that occurred (E1) during a specific time interval (E5), the desired event was present or not. Example of events that fall under this class:

- Forgetting to take medicines. In order to determine this, we need to know what events occurred (event types) along with their timestamps (E1) during a specific time interval (E5). From this we shall determine, if the desired event occurred or not.

While above we used events to determine if the desired event occurred or not, the combination of E1 and E5 can also be used to identify and avoid any untoward incident. For instance:

- Forgetting to close the main door after opening it
- Forgetting to turn off the iron box after turning it on

### C. Complex Event Detection

We formalise the identification of complex events using propositional calculus, which streamlines translating the events' condition-action into the Semantic Rules' first predicate logic [17]. In the list below, we exemplify the narration of each complex event or combination of events (Ei as explained above), and provide the formal propositional logic representation.

Event type (E1) - Open main door at midnight. Let  $t$  stand for the time at which the event occurs,  $e$  for the event type and  $c$  for the required complex event, then the condition that determines this complex event is:

$$[(e = \text{"Main Door Open"}) \wedge (0:00 \leq t \leq 8:00)] \rightarrow (c = \text{"Open Main Door at Night"})$$

Event type (E2) - Accidental fall inside bathroom. Here, we assume that a sensor capable of determining accidental fall as

an atomic event is available. Let  $e$  stand for the event type,  $l$  for the event location and  $c$  stand for the required complex event, then the condition that determines this complex event is:

$$[(e = \text{"Accidental Fall"}) \wedge (l = \text{"Bathroom"})] \rightarrow (c = \text{"Accidental Fall Inside Bathroom"})$$

Event type (E3, E4) - Frequent bathroom usage with or without high diabetes level. Let  $l_1, l_2, l_3$  be events that correspond to the "Entered Bathroom";  $t_1, t_2, t_3$  be the timestamps of these events;  $\epsilon$  be a fixed number, let  $h$  correspond to the historical patient data that says "High Diabetes Level" and let  $c$  be the required complex event, then:

$$[(e_1 = e_2 = e_3 = \dots = \text{"Entered Bathroom"}) \wedge (t_2 - t_1 < \epsilon) \wedge (t_3 - t_2 < \epsilon) \wedge (t_4 - t_3 < \epsilon) \dots \wedge (h = \text{"High Diabetes Level"})] \rightarrow (c = \text{"Frequent Bathroom Usage"})$$

Event type (E1, E5) - Forgetting to take medicine. Assuming the alarm needs to be raised when the patient forgets to take medicine at a particular time, then they need to be informed. Here, we need a special type of event, which is the passage of time. Let  $t_0$  stand for the current time and  $t_1$  and  $t_2$  represent the interval between which the patient should have had his meal and let  $e_1, e_2, e_3$  be the sequence of events that occurred between  $t_1$  and  $t_2$ , then the condition that determines this complex event is:

$$[(e_1 \neq \text{"Take medicine"} \wedge e_2 \neq \text{"Take medicine"} \wedge e_3 \neq \text{"Take medicine"} \dots \wedge (t_1 < t_0 < t_2))] \rightarrow (c = \text{"Forgot to Take medicine"})$$

### D. Complex events pre-processing

The aim of this section is to determine which complex events can be evaluated using the Semantic Web Rule Language SWRL rules in the ontology. As argued in our previous publication [6], SWRL satisfies the reasoning requirements for our use-case, and its rules can be encoded with the ontology for efficient, direct evaluation by the underlying semantic reasoning engine. Hence, our goal, where possible, is to hardwire the reasoning rules in the ontology. Otherwise, complex event pre-processor will be deployed that prepares/adapts the rules consumption by the semantic reasoning engine.

SWRL is a union of Horn Logic and OWL. Horn logic uses propositional calculus, and since we modelled our use-case events using propositional logic, this would aid in determine what can technically be implemented using SWRL rules. We need to be able to chronologically order the propositional variables either in order to determine the reoccurrence frequency of the same event or capture the ordered occurrence of a number of interdependent ones. For example, consider the complex event 3 (E3, E4) below:

$$[(e_1 = e_2 = e_3 = e_4 = \dots = \text{"Entered Room"}) \wedge (l_1 \neq l_2) \wedge (l_2 \neq l_3) \wedge (l_3 \neq l_4) \dots \wedge (t_2 - t_1 < \epsilon) \wedge (t_3 - t_2 < \epsilon) \wedge (t_4 - t_3 < \epsilon) \dots] \rightarrow (c = \text{"Restlessness"})$$

Here we need to be able to capture the difference of consecutive time intervals in Entering Room in order to check whether if it is less than a fixed value (threshold)  $[(t_2 - t_1 < \epsilon)$

$\wedge(t_3-t_2 < \epsilon) \wedge(t_4-t_3 < \epsilon) \dots]$  that might indicate Restlessness. We can store each event in the ontology along with an associated data type property that specifies the timestamp. Then in order to calculate the difference of consecutive time intervals, we have to first sort the events in chronological order before the difference can be calculated. However, there is no provision in SWRL for sorting the propositional variables. Hence, this sort of this complex event would have to be handled by a complex events' pre-processor at a preparatory stage preceding the semantic reasoning.

Similarly, we need to check whether the locations of any two consecutively occurring events are the same  $[(l_1 \neq l_2) \wedge (l_2 \neq l_3) \wedge (l_3 \neq l_4) \dots]$ . We can store the locations of events as data type properties associated with each event. As in the case above, we have to first sort the events before the required condition can be evaluated. Since there is no provision in SWRL for sorting propositional variables, this type of complex events will also need to be pre-processed.

Reasoning similarly, we can conclude that frequent bathroom usage should be handled at the pre-processor level.

## V. THE SEMANTIC DECISION SUPPORT FRAMEWORK

The framework's architecture is illustrated in Fig 3 below. The core part of our system, which is the ontology, contains all the taxonomy, the relations (object and data properties) and the reasoning rules. The semantic repository is accessed via a custom built Semantic Access API. All RDF triples are inserted into and accessed from the ontology by the pre-processor through this API. All that the pre-processor checks which events were atomic and which were complex before processing complex events and populating the ontology with both atomic / complex events (which can abstractly be thought of as alerts/triggers) into the ontology as RDF triples. The pre-processor is also responsible for invoking the reasoning engine and use the inferred triples to raise alarms such as `isRestlessness` and `alarming isFrequentBathroomUsage`, which are inferred based on the specific patient's context.

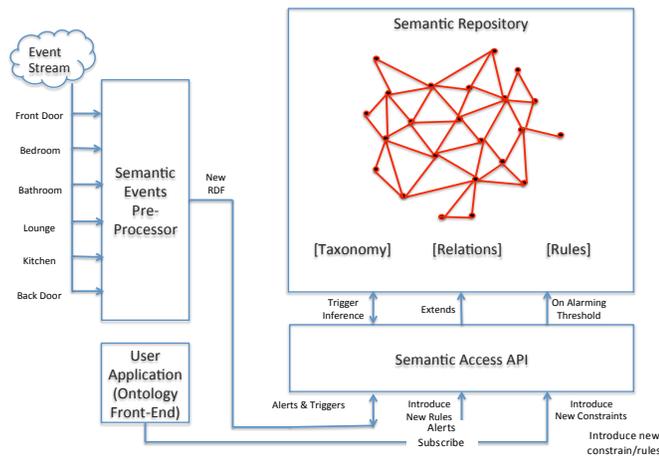


Fig. 3. System Framework

The input to the pre-processor is an event stream, which in our use-case is mainly composed of the patients' room occupancy data. As outlined in the previous section, initially

the events are represented using propositional logic in order to identify those parts of these events that need to be processed at the pre-processor-level as outlined in the previous section. This solution is not just simple but is also mathematically precise.

The user front-end serves two purposes, updating the reasoning engine with new rules/constraints, and subscribing to certain alerts from the Dementia care decision support system.

### A. The dynamics of the complex events ontology

A detailed view of the 'Events' ontology class is shown in Fig 4. The class comprises two event sub-classes, 'AtomicEvents' and 'ComplexEvents'. These classes in turn have further sub-classes corresponding to various types of events. The reason for giving names like 'AtomicEvents' and 'ComplexEvents' is to demonstrate how to create the ontology for Semantic Complex Event Processing. Real-life project would have more realistic names. All atomic events are inserted as individuals of the sub-classes of the 'AtomicEvent'. There is no additional processing performed by the pre-processor on these atomic events. Atomic events are inserted into the ontology as and when the pre-processor receives them. However, those events that cannot be calculated by SWRL rules (complex events) are processed by the pre-processor and inserted as individuals of the sub-classes of the class 'ComplexEvents'. Once all the individuals are inserted, the classification in the ontology is solely performed by the rules.

An important subclass of events is 'TimePassageEvent'. This class has a single individual 'i\_CurrentTime', which stores the current time through the data type property 'hasDP\_Time'. The current time is populated by the pre-processor and is used by the semantic rules to classify some of the events. This is important for time-dependant events such as prolonged stay in a specific room.

During classification, the rules use each patient's medical history as well for classifying the atomic / complex events. Each patient's medical history is stored in the second section, which is the class 'PatientHistory'.

The other subclass is 'AlarmingEvents'. All other classes are internal to the ontology and required only by the rules. This is the only class required by the external application making use of the classification. This class has various subclasses, which are used to insert the asserted triples. Thus there is a subclass called 'ForgotToTurnOffGas'. If an event instance comes under this class then that the event instance is an alarming event and means the patient associated with the event forgot to turn off gas.

The Patient class enables the ontology to be multi-patient. All instances of this class are patients and every event corresponds to a unique patient by using object properties. Thus an event called 'i\_Door\_SuganthRamaswamy' may be associated with the patient 'i\_SuganthRamaswamy' through the object property 'hasOP\_EventDoor'.

Finally, the 'System' class has a single instance 'i\_System', which stores a single data type property that specifies the update frequency. The pre-processor uses the value of this property to set the frequency of its timer variable, which in turn

determines how often the pre-processor populates the ontology and invokes the reasoning engine.

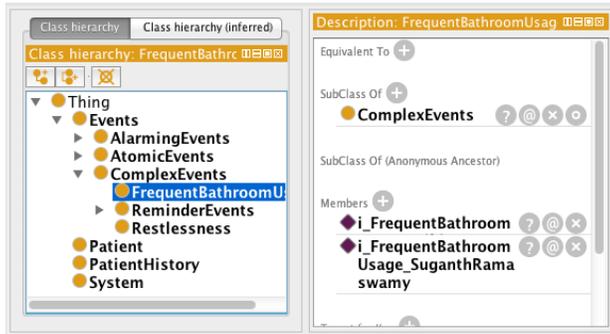


Fig. 4. Events ontology top-level view

### B. Semantic rules for the decision support process

The core reasoning activity of the DCDSS was implemented using SWRL. Below we exemplify how the propositional logic statement representing ‘prolonged use of a room’ was translated into a SWRL Rule.

$$[(e_1 = \text{"Entered Room"}) \wedge (l_1 = \text{"Bathroom"}) \wedge (t_0 - t_1 > \epsilon)] \rightarrow (c = \text{"Forgot to Close Main Door"})$$

The above propositional statement is translated into a SWRL rule as follows:

```
EnterRoom(?er), TimePassageEvent(?tp), hasDP_Location(?er, ?erl),
hasDP_Time(?er, ?ert), hasDP_Time(?tp, ?tpt), greaterThanOrEqual(?diff,
8.0), stringEqualIgnoreCase(?erl, "Bedroom"), subtract(?diff, ?tpt, ?ert) ->
ProlongedUsageOfBedroom(?er)
```

Below is a functional description of the above rule:

1. EnterRoom(?er): Retrieve all room events and store them in the variable ‘er’
2. TimePassageEvent(?tp): Retrieve the special time passage event and store it in the variable ‘tp’
3. hasDP\_Location(?er, ?erl): Retrieve the location of the room events and store them in the variable ‘erl’
4. hasDP\_Time(?er, ?ert): Retrieve the time of the room events
5. hasDP\_Time(?tp, ?tpt): Retrieve the current time
6. subtract(?diff, ?tpt, ?ert): Store the difference between the current time and the time of room events in the variable ‘diff’
7. greaterThanOrEqual(?diff, 8.0): Check if the difference calculated above is greater than or equal to 8 hours
8. stringEqualIgnoreCase(?erl, "Bedroom"): Also check if the location of the event is bathroom
9. ProlongedUsageOfBedroom(?er): If all of the above conditions match then, classify the events as prolonged usage of bathroom

Example of Further SWRL rules that were integrated into the ontology are illustrated in Fig 5 below. It is worth noting that some of the events are simply classified without any further processing. These are complex events that are

calculated and populated in the ontology by the pre-processor. Although the pre-processor could have placed them under AlarmingEvents directly, the solution is not elegant. For example, in the future if we had to do some further processing on these events, then the approach that has been taken here would result in minimal modification to the ontology.

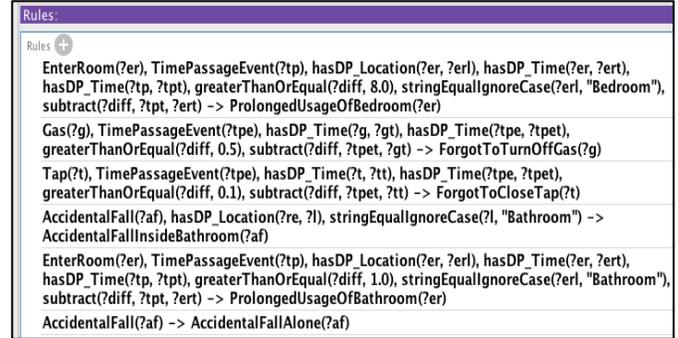


Fig. 5. SWRL Rules View

## VI. EXPERIMENTATION AND EVALUATION

### A. Experimental Setup

Jena [18] was the API of choice for accessing the Semantic repository as it has built-in facility to execute SWRL rules. Jena is supported by an open-source RDF repository, Jena TDB [19] that despite lagging in performance in comparison with other commercial RDF repositories [20], its RDF storage and query capabilities are adequate for the complexity of our use-case, in addition to the critical factor of full support for SWRL. The experiments were carried out on a Mac Mini workstation, with 2.4 GHz Intel Core 2 Duo processor and 8GB DDR3 memory.

### B. Evaluation Criteria

The response times for various numbers of patients were gathered in the previous phase, which were then analysed during the experimentation, and the evaluation phase. We mainly evaluated the systems response time, taking into consideration the knowledgebase reasoner warm-up time. In each case, experiments were carried out for a number of patients between 10-120, which is estimated to be satisfactory for our DCDSS. Five different samples were collected and their averages were taken to perform the analysis.

### C. Statistical Data Analysis

It was important to evaluate the results obtained using a statistical data analysis tool in order to verify the significance of the results derive an equation to predict the response-time of the system if the number of patients in the system changed via regression analysis methods.

Minitab, a popular statistical software program that provides a wide range of basic and advanced capabilities for statistical analysis [21], which is capable of evaluating the significance of the results obtained as well perform regression analysis.

In statistics, linear regression analysis is a technique that is widely used for prediction and forecasting. Linear regressions analysis helps one to understand how the dependent variable in an equation varies if one or more of the independent variables change. The equation we are talking about is essentially a linear equation that matches the graph obtained in the closest manner possible.

Following is some terminology related to regression analysis:

- p-value: This is the significance value that says that if it is less than 0.05, then the results are significant.
- Adjusted R-Squared value: The adjusted R-squared value describes how closely the straight-line represented by the linear equation matches the graph. The closer it is to 100%, the closer the match is.

In all our tests, the p-value was 0.000 on Minitab, which means that the results were all significant, and the adjusted R-Squared values were all either greater than 85% or were greater than 90%.

#### D. Response Time

This represents the response time without including the warm-up time. Every time the reasoning engine is invoked for the first time, the time taken to reason is higher than the time taken to reason in the later invocations so long as the connection to the semantic repository is not closed. It is important to determine the response time sans warm-up time because many applications load the repository once and then invoke the reasoning engine over and over again without closing the connection to the repository. When applications invoke the reasoning engine several tens or hundreds of times, the response time will approximately average out to the one without warm-up time.

The table below shows the response time against the number of patients. Response time is the time taken to execute the rules and classify the individuals in the ontology.

TABLE I. RESPONSE-TIME

Number of Patients	Response-Time (msec)
10	348.4
20	476.6
30	733
40	724.2
50	885.4
60	1065.2
70	1463.8
80	1232
90	1429.8
100	1871.8
110	2118.4
120	2620

The output from Minitab for the above data is shown in Fig 6 below. The p-value obtained was 0.000, which means the results were significant. The linear equation that we got for the response time against the number of patients was:

$$Response\ Time = 49.6 + 18.43 * Number\ of\ Patients$$

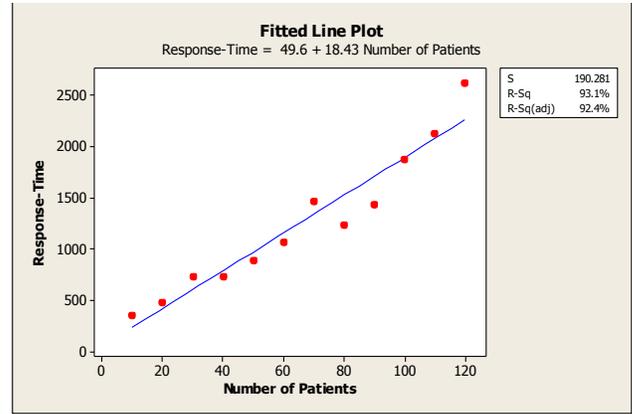


Fig. 6. Response-Time

#### E. Response time at first run

For completeness, we measured the response time the first time the reasoning engine is invoked, which is predictably significantly higher than subsequent invocations. This is important as in certain situations it may not be possible to keep the connection to the repository opened once and then invoke the reasoning engine over and over again. This can manifest if patient monitoring deploys a web-based connection (as opposed to dedicated, stream-based connection) to the decision support system, where the reasoning engine is invoked via a web-based method that is normally stateless. Every time the web method is called, the connection would have to be opened in the beginning and be closed before the web-method returns.

The response time upon the first invocation of the reasoning engine is recorded in the table below.

TABLE II. RESPONSE TIME FOR 1ST INVOCATION OF REASONING ENGINE

Number of Patients	Response Time (msec)
10	698
20	1590
30	2334
40	2686
50	4824
60	3995
70	4812
80	4026
90	5701
100	5197
110	5579
120	6696

The output from Minitab for the above data is shown in Fig 7, where the p-value obtained was 0.000, which means the results were significant. The linear equation that we obtained for the warm-up time was as follows:

$$Response\ Time = 933.8 + 47.35 * Number\ of\ Patients$$

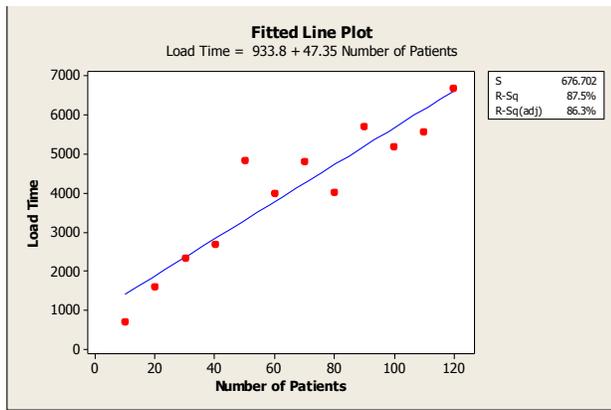


Fig. 7. Response time upon first invocation of the reasoning engine

## VII. CONCLUSION AND FUTURE WORK

The motivation for the research documented in this paper is that Semantic Web technologies can play a useful role in building Dementia care decision support systems by intelligently interrelating the abnormalities in patient behaviour captured by sensory devices to Dementia symptoms prescribed by clinical guidelines in order to assist medical advice. Our semantic-based approach integrates information about related sensory events, patient medical history, and Dementia symptoms into a semantic ontology that uses a rule-based system to infer knowledge about the Dementia patient medical state. We extensively investigate the role of semantic-based knowledge inference in remote healthcare decision support systems and contribute to the understanding of the required computational capacity, reasoning complexity, and interactivity dynamics of the semantic engine with the sensory equipment.

Our investigation establishes that handling the interrelated complex events emanating from a Dementia patient monitoring system is challenging for semantic-based decision support, and concludes that both the basic semantic rules systems such as SWRL and the semantic complex event processing languages such as EP-SPARQL only partially fulfil the requirements of the intended remote healthcare decision support; hence, we introduced a new approach that relies on propositional calculus modelling to segregate complex events that are amenable for semantic reasoning from events that require pre-processing outside the semantic engine before they can be reasoned upon. The event pre-processing activity also controls the timing of triggering the reasoning process in order to further improve the efficiency of the inference process. The performance of the semantic-based decision support system is assessed using regression analysis, where the response-time is evaluated against the number of monitored patients, and it was concluded that the incurred overhead by the semantic reasoning process is tolerable within the context of Dementia care decision support.

Our future work will focus on facilitating expert-free extension of the semantic ontology with new atomic/complex events and update/modification of the associated inference rules.

## REFERENCES

- [1] M. Strachan, J. Prince, and B. Frier, "Diabetes, cognitive impairment, and Dementia", *BMJ* 2008 336:6, pp. 2928-2935.
- [2] C. Thompson, K. Spilsbury, J. Hall, Y. Birks, C. Barnes, and J. Adamson, "Systematic review of information and support interventions for caregivers of people with Dementia", *BMC Geriatrics* 7: 18.
- [3] G. Young, "The implications of an ageing population on the UK economy", Bank of England Working Paper, available at: <https://www.bankofengland.co.uk/publications/workingpapers/wp159.pdf> (last accessed: 6/03/14)
- [4] "Just Checking Dementia support system", <http://www.justchecking.co.uk/> [accessed: 6/3/2014]
- [5] F. Meiland et al., "COGKNOW: Development and evaluation of an ICT-device for people", in: *Medical and Care Compunctics* 4, pp. 166-177.
- [6] T. Osman, S. Rmaswamy, S. Mahmoud, and M. Saeed, "Utilising Semantic Technologies for Decision Support in Dementia Care", in proceedings of the 15th International Conference on Computer Modelling and Simulation-UKSim 2013, pp. 628-633.
- [7] I. Horrocks et al., "SWRL: A Semantic Web rule language combining OWL and RuleML." W3C Member submission 21 (2004): 79.
- [8] A. Lotfi, C. Langensiepen, S. M. Mahmoud, and M. J. Akhlaghinia, "Smart homes for the elderly Dementia sufferers: Identification and prediction of abnormal behavior", *Journal of Ambient Intelligence and Humanised Computing*, Sept 2012, Vol. 3, Issue 3, pp. 205-218.
- [9] A. Kenner, "Securing the elderly body: Dementia, surveillance, and the politics of aging in place", *Surveillance and Society*, Vol. 5, No3, 2008, pp. 252-269.
- [10] S. Mahmoud, A. Lotfi, and C. Langensiepen, "User Activities Outlier Detection System Using Principal Component Analysis and Fuzzy Rule-Based System", Fifth International Conference on Pervasive Technologies Related to Assistive Environments, Crete, Greece, 2012, Article No 26.
- [11] S. Yunmei, et al., "Semantic-based data integration model applied to heterogeneous medical information system", In the 2<sup>nd</sup> International Conference on Computer and Automation Engineering (ICCAE), Vol. 2.
- [12] A. Gómez-Pérez, M. Fernández-López, and Oscar Corcho, "Ontological engineering: with examples from the areas of knowledge management, e-commerce and the Semantic Web", *Advanced Information and Knowledge Processing*, Springer 2004.
- [13] T. Osman, D. Thakker, and M. Nathan, "Research Notes on the Practical Deployment of Semantic Knowledge Bases", Proceedings of the 12th European Conference on Knowledge Management ECKM 2011 - Passau, Germany. Vol.1 pp. 737-745.
- [14] Alzheimer's Association, 2010 Alzheimer's Disease Facts and Figures, *Alzheimer's & Dementia*, Volume 6. (2010)
- [15] K. Teymourian, O. Streibel, and A. Paschke, "Towards Semantic Event-Driven Systems", *New Technologies, Mobility and Security (NTMS)*, 2009 3rd International Conference, pp 1-6.
- [16] D. Anicic, P. Fodor, S. Rudolph, and N. Stojanovic, "EP-SPARQL: A Unified Language for Event Processing and Stream Reasoning", Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Seiten, pp. 635-644.
- [17] SemanticWeb.org, "Propositional Calculus Wiki" [online], available at: [http://semanticweb.org/wiki/Propositional\\_calculus](http://semanticweb.org/wiki/Propositional_calculus), last accessed 7/3/14.
- [18] Jena 2 Inference Support, <http://jena.sourceforge.net/inference/> (last accessed 3/3/2012)
- [19] TDB - a component of Jena for RDF storage and query. Available at : <http://jena.apache.org/documentation/tdb/>. [accessed on 17/3/2014].
- [20] D. Thakker, T. Osman, S. Gohil, and P. Lakin, "A Pragmatic Approach to Semantic Repositories Benchmarking", *The Semantic Web: Research and Applications*, LNCS, Vol. 6088, 2010, pp 379-393.
- [21] BF Ryan, BL Joiner, J Cryer, 2005. Minitab handbook: Updated for release 14. Fifth Edition. Brooks/Cole, pp. 1-7
- [22] D. Celik et. al, "Mobile Pediatric Consultation and Monitoring System through Semantic Web Technology", in the 38th International Conference on Computer Software and Applications Conference Workshops (COMPSACW), July 2014, pp. IEEE: 354 - 359.