Fall 2013

Application of Self-Monitoring for Situational Awareness

Christopher Trickler

APPLICATION OF SELF-MONITORING FOR SITUATIONAL AWARENESS

by

CHRISTOPHER TRICKLER

(Under the Direction of Vladan Jovanovic)

ABSTRACT

Self-monitoring devices and services are used for physical wellness, personal tracking and self-improvement. These individual devices and services can only provide information based on what they can measure directly or historically without an intermediate system. This paper proposes a self-monitoring system to perform situational awareness which may extend into providing insight into predictable behaviors. Knowing an individual's current state and likelihood of particular behaviors occurring is a general solution. This knowledge-based solution derived from sensory data has many applications. The proposed system could monitor current individual situational status, automatically provide personal status as it changes, aid personal improvement, contribute to other self-monitoring systems, and enhance other life-tracking objectives.

INDEX WORDS: Self-Monitoring, Quantified Self, Data Mining, Sensor Fusion, Situational Awareness, Predictive Analytics, Behavior Monitoring

APPLICATION OF SELF-MONITORING FOR SITUATIONAL AWARENESS

by

CHRISTOPHER TRICKLER

B.S., Georgia Southern University, 2009

A Dissertation Submitted to the Graduate Faculty of Georgia Southern University in Partial Fulfillment of the Requirements for the Degree

MASTER OF COMPUTER SCIENCE

STATESBORO, GEORGIA

2013

APPLICATION OF SELF-MONITORING FOR SITUATIONAL AWARENESS

by

CHRISTOPHER TRICKLER

Major Professor:	Vladan Jovanovic
Committee:	James Harris
	Wen-Ran Zhang

Electronic Version Approved:

December 2013

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Self-Monitoring Technology

Self-monitoring through technology has led to a movement known as Quantified Self. The Quantified Self movement [20] was in part created by Gary Wolf and Kevin Kelly in an effort to collect and document self-monitoring and tracking tools [30]. The Quantified Self or self-tracking is the practice of using technology to track personal status over time for informative uses and self-improvement as an example. Traditionally, self-monitoring has been limited in devices available or software services and tools. Product manufacturers will often have their own proprietary repository of sensor data collected from their own products.

There are some web-based services which have emerged allowing for aggregation of sensor data from multiple vendors such as Microsoft Health Vault [13]. However, these services are not guaranteed to be available; a prime example is that Google Health [4] was cancelled. The information available from these proprietary repositories contains only basic trends of data for specific data types. A common correlation among fitness-centric sites[3] is the number of steps taken and calories burned leading to expected weight loss. The capabilities of proprietary data format systems as an intermediate system is limited. The uncommon process of discovering relations between various types of sensor data from multiple vendors is one of them. Remote locations of information storage resulting in potentially long latencies for sensor processing are another limitation. The system proposed in this paper addresses a few of the aforementioned problems.

The design of the system will use sensor data from multiple types of devices and services originating from multiple manufacturers, but aggregated and maintained central to an individual.

Aided by a classifying system [27] to determine product capabilities, a diverse collection of devices and services were chosen for testing and evaluation of the system, see figure 11. The self-monitored data could then be transformed into meaningful data points useful for analyzing trends, detecting patterns in behavior, and knowing the status of nearby environment. Lastly, the resulting status and trends is presented to intended recipients. In summary, the proposed system will collect sensor data and apply data classification, mining and analysis techniques: resulting in current status with known past, present, and likely behaviors.

History

Self-monitoring, activity tracking, life-tracking, self-quantification are all essentially the same processes. The processes, or technology utilized for that matter, are not new. Physical trainers have been monitoring athletes' performance for many years using paper records and basic trend analysis. Social scientists have studied behavioral patterns as well using similar methods. The act of self-monitoring through technology has improved over the years, emerging into the consumer market. The progress of consumer sensory products and reduced price of product design are resulting from various high-fidelity but low-cost sensory components. These advances are opening opportunities for improved tracking devices and services.

Typical activity classification systems use raw sensor data processing techniques to generate a context or activity to some degree of certainty [19], [24]. The raw sensor data is often collected from custom, not commercial, devices created specifically for research and experimentation. Within these specialized systems, proprietary data types may exist and be stored within the hardware's' memory or own information repository. Research regarding alternative sensor data processing and activity classification is promising. Eventually, these concepts and ideas could translate into consumer products.

Innovation

Previous research as indicated earlier has utilized custom sensory tracking processes. The design of this system is intended for integration with existing consumer, commercial, or custom sensory devices and services. Configuration of processes or components within the system must be allowed. Rather than supporting a fixed set of recognizable activities or limited collection of data types the system should be scalable. Contrast this with other activity tracking systems which may use many custom hardware solutions, none of which are consumer friendly. Also, comparison of other systems may not allow access to historical data, only daily totals. The intended design of the system must also consider various sampling frequencies or availability of data. Conversion of data from similar sensor data types, formats, and units may seem trivial but can be computationally expensive.

Mainstream self-monitoring systems may not allow custom data analytics, custom trend reports, or collection of sensor data from multiple sources. In regards to sensor data, many services do not evaluate data for complex trends or relations, only product and domain specific knowledge. Hindrances such as limited querying of data, license agreements of web services, and communications access latency all affect system capabilities. In order to accurately represent situational awareness and monitor behavior progression within a reasonable period of time much of the processing needs to be localized. It is therefore desirable to maintain data relatively close to where key sensory devices and services are available such as a mobile device or computing device within a common location. Using the aforementioned requirements for innovation will theoretically yield an extendable system for situational awareness and behavior detection.

CHAPTER 2

SYSTEM DESIGN

System Objectives

The principle goal of the proposed system design is to enable situational awareness functionality and predictive behavior analytics to an individual. Situational awareness is a collection of current meaningful sensory data. Intelligent data is information which has been processed for human readability. Rather than expressing raw sensor data values such as 75 degrees Fahrenheit, the intelligent interpretation of that data for an individual may be ‘Hot’.

Analyzing temporal, or time-based, sensor data for patterns may yield predictable behaviors. Potentially, patterns describing behaviors can be detected within reoccurring changes in situational state; for instance transitioning from ‘Hot’ to ‘Cold’ to ‘Hold’ to ‘Cold’ again. This pattern recognition may then be analyzed to determine a more complex behavior such as ‘Getting Tools’ in the case of a building contractor entering or leaving a building to get work related tools. In this example, not only would the location provide a relation to the individuals’ behavior, but the temperature as well (due to varying temperatures in the area).

The knowledge obtained from discovering predictable behaviors could be used for continuous improvement, the main strategy [7] of which is improving processes and eliminating waste. In the previous example, reviewing a work day that exhibits multiple instances of ‘Getting Tools’ could lead to an insight that moving tools closer to the work location would save time.

System Overview

A general process for using sensory data to build useful situational status and perform behavior monitoring follows. At first a general overview is described, followed by individual components with further details. The collection, analysis, and presentation of behavioral patterns and situational status are treated together as a system. Initially, self-monitored or personal sensory data is acquired automatically or manually before it enters the system. In figure 1, the components from which sensor data is collected and grouped together will be known as providers. Providers may be websites, portable devices, stationary devices, or other sources of sensory data including manual data entry. The sensor data collected is stored, filtered and analyzed within a system component known as the engine. The engine is a continually running process. Within the engine, data is analyzed for patterns appropriate to operational parameters and solutions are generated as a result of the process. The solutions are in the form of current status and patterns of data in user-readable format. The solutions are the end result of a complete process from receiving sensor data to analyzing patterns; they are the current situational status and probabilistic behaviors.

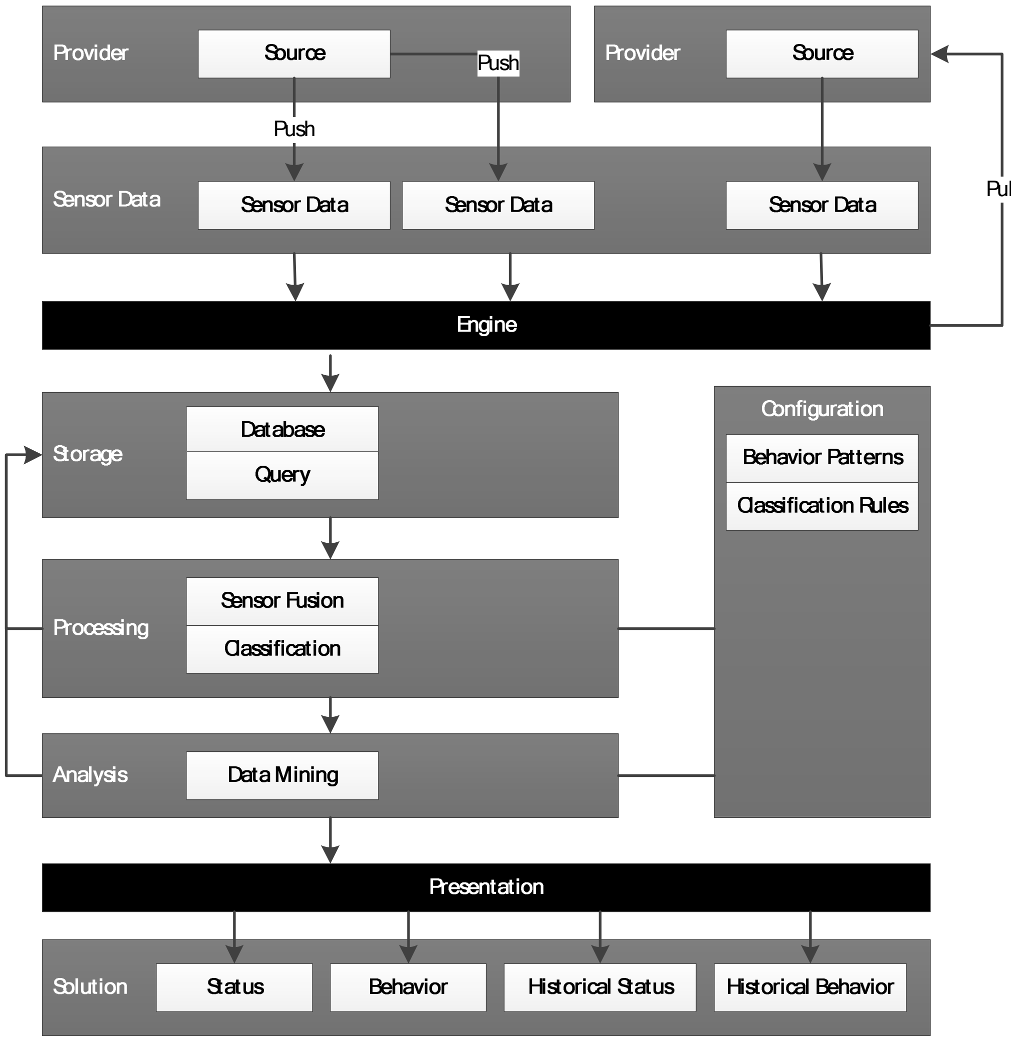


Figure 1. System overview of components and processes.

CHAPTER 3

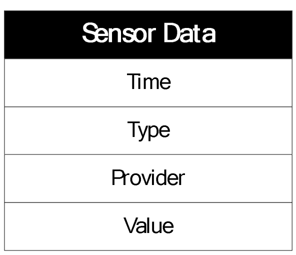
SYSTEM OPERATION

Sensor Data

Sensory data is the primary input to the system and can be acquired by automatic or manual collection. Sensor data can be pushed or pulled at different rates such as near real-time, once a minute, every hour or other interval. Utilities like data scrubbers [16], or database tools may be required to get access to raw unprocessed data within a website or other storage media. As illustrated in figure 2, the sensor data rate can vary based on sensor data type. Additionally, the device or service and type of data could determine the data rate; recording a person's weight every hour is not needed. Conversely, recording a person's location too slow could result in missed status changes. The source of the sensory data may vary from websites, mobile phones, applications, or any specifically which may monitor personal data. The types of sensory data available also depend on the device. There are many types of sensory data including, but not limited to: steps taken, compass heading, global positioning, time, temperature, weight, and others.

Each type of sensory data measurement shares common attributes as modeled in figure 2. These common attributes are: the time at which the measurement was taken, sensor data type, source of the data, and a sensor value. As previously hinted at, the data will be aggregated within a database, and after processing will become more meaningful than its raw form in this stage. Pre-processing however could be handled at this stage, specifically within the sensor provider. An example of local processing is translating analog readings of an accelerometer within a mobile device to a meaningful type: movement. Sensor data from multiple sources may also require unit conversions to standardize the data values. Unit conversion as a pre-processing step within the provider is recommended to reduce overall system processing, but this may not be possible due to device or service accessibility.

Raw sensory data analysis is beyond the scope of this system but is studied in detail by many subject experts. Sensory data as it relates to human health for instance [17] could be analyzed at the device level for converting kinematic motion data to bodily posture detection. A single raw sensor value could therefore result in one or more types of sensor data after processing potentially having additional meaning. Many of the examples or equipment used in the testing of this design makes use of existing interpretations of sensory data or a post-processing step.



Sensor Data			
Provider	**Sensor Type**	**Sensor Value**	**Rate**
Fitness	Movement	0 Steps	1 / Minute
Fitness	Weight	210 Pounds	1 / Day
Fitness	Weight Timestamp	05/01/2013 5:35 A.M.	1 / Day
Mobile	Location	33.9531° N, 83.9925° W	1 / Minute
Mobile	Heading	45°	1 / Minute
Mobile	Movement	Idle	1 / Minute
Mobile	Time	05/02/2013 5:40 A.M.	1 / Minute
Custom 1	Weather	Light Rain	1 / Hour
Custom 1	Temperature	45° Fahrenheit	1 / Hour
Custom 2	Application	Document1.doc	1 / Minute
Custom 2	Cursor	Active	1 / Minute
Custom 3	Thought	Coffee	1 / Hour

Figure 2. Sensor data model and sample raw sensory data.

Configuration

Recognition of behaviors for use in behavior tracking or situational awareness benefits from having an existing knowledgebase to compare patterns of data to. A configuration data set is used to populate the engine classification and data mining components for aiding in detecting patterns of data. Methods could be used to learn patterns as they occur via user-feedback such as by supervised learning techniques. However, having a known static list simplifies the analysis for verifying results within this system. The configuration could be a collection of files or data within the database which is associated with an individual using the system. A configuration per user would speed identification of patterns for a given user, but restrict number of patterns capable of being recognized overall. In this system the configuration is designed for a single user and therefore can only recognize behaviors of an individual. This decision was made in part due to the individual preferences for a given user.

A single person has particular interpretation of sensory data and their own individual daily routines. The methods applied for classification of sensory data is therefore made a part of the configuration dataset. Figure 3 shows a partial configuration of the classification process for a user. The configuration dataset is also composed of a list of behaviors capable of being detected: each behavior is a collection of patterns of temporally sequential data. The patterns contain members with attributes that represent state, duration, repetition, and order. For every sensor data type within a behavior there is one pattern. In the example figure 4, the behavior ‘Playing Sport’ has two patterns of data which must match to be considered. One pattern is recognized when sensor data representing the ‘Movement’ data type are either ‘Walking’ or ‘Idle’ for 1 or more minutes. Another sequence in the behavior is a ‘Location’ data type which must have the value of ‘Park’ for duration of 45 or more minutes.

Ideally, the sequences would be represented with syntax similar to regular expression parsers [26] to enable fast, token-based recognition. A declarative language such as Prolog [25] could also be used for a more complex and relational programming-based approach. The choice of syntax for the configuration set impacts the process of mining known patterns. In the format described below the sensor type is given for a pattern with tuples representing the value and time duration (or range) within brackets. If a behavior has more than one pattern that must be recognized it is listed as well with its respective sensor data type.

Classification Configuration			
Sensor Type	**Class Type**	**Class Definition**	**Data Classes**
Movement	Range	0 – 60 Steps	Idle
		61 – 100 Steps	Walking
		101 – 150 Steps	Jogging
		151+ Steps	Running
Weight	Range	141 – 160 Pounds	Underweight
		161 – 175 Pounds	Ideal
		176 – 225 Pounds	Overweight
Temperature	Range	0° – 32° Fahrenheit	Freezing
		33° – 65° Fahrenheit	Very Cold
		66° – 75° Fahrenheit	Mild
		76° – 85° Fahrenheit	Hot
		86° – 95° Fahrenheit	Very Hot
Application	Text	{Thesis.doc, *Sensor Fusion*}	Thesis Research
		{*Penny-Arcade*, IMDB, Slashdot}	Wasting Time
Location	Spatial	S(current, home)	Home
		S(current, office)	Office
		S(current, garage)	Garage
		S(current, park)	Park
		S(current, store)	Store
		S(current, work)	Work

Figure 3. Configured classification data set.

Behavior Configuration
Name
Types[]
Patterns[]

Behavior Configuration		
Behavior	**Sensor Type**	**Pattern Definition**
Grocery Shopping	Location	[Store, 5+ Minutes]
Writing Paper	Application	[Thesis Research, 30+ Minutes
	Cursor	[Active, 10+ Minutes]
	Location	[Home, 30+ Minutes]
Playing Sport	Location	[Park, 45+ Minutes]
	Movement	[Idle, 1 – 2 Minutes],[Walk, 1+ Minutes]
At Work, Taking Break	Location	[Work, 5+ Minutes]
	Movement	[Walk, 2+ Minutes]
	Application	[Wasting Time, 1+ Minutes]
At Work, Coding	Location	[Work, 5+ Minutes]
	Movement	[Idle, 2+ Minutes]

Figure 4. Configured behaviors data set.

Database

Sensor data is collected and stored within a database which is not necessary to specify for this system. Multiple types of database solutions exist from multiple vendors [12],[14] each with their own capabilities. The database component of this system is required due to numerous elements of data stored over time which need to be queried based on unique properties. However, an in-memory database would be sufficient. Utilizing a database allows for: querying of specific data elements quickly through indexing, replication of data to a remote site, and distribution of data processing.

An in-memory database would be ideal due to potentially high amounts of record creation and deletion during an aggregation process described later known as sensor fusion. The database requirements are relatively simple in regards to the features and data models implemented within this system. A more complex system designed for many users, security constraints, profiles, nationalization, and additional features would require more specific database requirements. As a continually operating process, the sensor data is stored and analyzed, possibly generating additional data or removing specific elements. The process is commonly described as information fusion or more specifically in this system as sensor fusion.

Sensor Fusion

Sensor fusion is the process of obtaining qualitative data from often unreliable or multiple sensor sources. The process of sensor fusion may be applied within the device or service domain, a de-centralized approach with respect to source of data. Sensor fusion is also used in this proposed system within the engine, a centralized approach. Many data fusion techniques exist [23], their purpose being relatively common: quality data from an unreliable source. There are several levels of data fusion [8] of which only the Level 0 and Level 1, signal refinement and object refinement respectively, are used [21].

Described earlier was a process of collecting sensory data from multiple sources, the providers. Frequently, the same type of data can be obtained from different sources. Different sources of data such as location information could be obtained from global positioning sensors, inertial sensors and pedometers. Each of these sources could provide similar but not identical, or even conflicting states. Conversely, sensory data sources could provide identical states in which case it would be beneficial to eliminate duplicate data such that fewer items have to be analyzed later in the system.

The strategy for performing sensor fusion can differ based on sensor data type. In some scenarios it is beneficial to use decentralized data fusion for the purpose of reducing information loss [18] such as geo-positioning coordinates. In other cases, like computationally expensive tasks, it is better to allow the system to manage the data fusion as a centralized process. The sensor fusion process merges similar sensor data and removes duplicate data. If the sensor data is considered similar or not having a significant change over time the results are combined into a single fused sensor data element. The similarity measurement [6] can differ for every sensor data type by having different ranges, thresholds, or rules. For example, similarity in measuring a floating point global-positioning data value could be different than an integer based heading data value. After comparing similarity and qualifying to be fused, the single fused sensor data element has almost identical attributes as the standard sensor data.

Fused sensor data no longer requires the provider, due to multiple providers potentially contributing to the same type of sensor data. Having both a mobile phone and a separate digital compass for instance is useful in having redundant sensory data, but not required to know the source of the data in this system. Additionally, the fused sensor data has an added attribute representing length of time or duration, detailed in figure 5. The timestamp and duration of the fused sensor data allows for knowing when and how long the state of a sensor has existed within a specific state. When performing data classification and mining operations it will become apparent the need to know how long a sensor was in a given state. In regards to the database component of the system, performing sensor fusion also allows for the combination of data thus removing duplicate or similar data reducing storage requirements and length of processing time.

Fused Sensor Data
Time
Type
Duration
Value

Sensor Data			
Time	**Type**	**Provider**	**Value**
1:00 PM	Movement	Mobile	Idle
1:02 PM	Movement	Mobile	Idle
1:05 PM	Movement	Fitness	Run

Fused Sensor Data			
Time	**Type**	**Duration**	**Value**
1:00 PM	Movement	5 Minutes	Idle
1:05 PM	Movement	0 Minutes	Run

Figure 5. Fused sensor data model and sample sensory data.

Classification

Classification is a process for separating data into categories or classes of data. The classification process is defined by the application-specific requirements and attributes of data to be classified. Classification systems [6] can organize data by categories defining shape, color, specific enumeration, common value, or other parameterized groupings. The classification strategy used within this system is chosen based on the fused sensor data type whether to use a decision tree, textual class, numerical range, spatial class, closest distance, or other methods. Some of the configured data classes and their respective matching criteria are shown in figure 3.

One of the common examples used within this text is a fused data item with the ‘Location’ type. This positional sensor data originated from raw geo-positioning coordinates: latitude and longitude. The classifying process examines the coordinates from the raw sensor data value and calculates the distance to known locations. A database of indexed areas is consulted to determine location sensor data class based on closest distance match between considered location and known locations. Typically, one of many distance functions [6] relating one data value to another is used to compare and match data into a given class. This process reduces future computation by updating the sensor value with the new classified value. The classified value replaces a fused sensor data item with another one containing more meaningful data: ‘Home’ for instance rather than the raw geo-positioning coordinates. Because data classification is effectively Level 1 data fusion (object refinement), the fused sensor type becomes the classified value.

<table>
<tr><th colspan="6">Fused Sensor Data</th></tr>
<tr><th>Type</th><th>1301</th><th>1302</th><th>1303</th><th>1304</th><th>1305</th></tr>
<tr><td>Movement</td><td colspan="3">61</td><td colspan="2">152</td></tr>
<tr><td>Heading</td><td colspan="2">45°</td><td colspan="3">86°</td></tr>
<tr><td>Weather</td><td colspan="5">Light Rain</td></tr>
<tr><td>Application</td><td colspan="3">‘Trends in Sensor Fusion’</td><td colspan="2">‘’</td></tr>
<tr><td>Location</td><td colspan="3">33.9531° N, 83.9925° W</td><td colspan="2">33.6664° N, 84.0075° W</td></tr>
<tr><th colspan="6">Classified Sensor Data</th></tr>
<tr><td>Movement</td><td colspan="3">Idle</td><td colspan="2">Run</td></tr>
<tr><td>Heading</td><td colspan="2">North East</td><td colspan="3">East</td></tr>
<tr><td>Weather</td><td colspan="5">Light Rain</td></tr>
<tr><td>Application</td><td colspan="3">Thesis Research</td><td colspan="2"></td></tr>
<tr><td>Location</td><td colspan="3">Home</td><td colspan="2">Work</td></tr>
</table>

Figure 6. Classification of fused sensor data example.

Data Mining

Temporal data mining [5] is utilized in this system as a method of pattern discovery and determining the likelihood of recognizable behaviors. The discovery process occurs after classification because it requires meaningful data, not raw sensor data, to find useful patterns. The patterns of data, or behaviors, have been defined previously as part of the configuration of the engine. The extents or total window of time for which the engine recognizes patterns works off of a 24 hour schedule such that it tracks daily behaviors. By limiting the time to be scanned for patterns, the start of the scanning begins with the start of the day and ends with the last minute of a given day. However, the activity window may be adjusted for an individual's lifestyle.

The data mining algorithms used are based on finding sequences or reoccurring patterns of data over time. For time based data a temporal data mining method such as serial sequence matching [5] is used to locate the recognizable behaviors. Serial sequence matching organizes the data by time or order in which the event occurs per data type classification. In this system, the sequences are generated based on the time per classified sensor data type. The serial sequences are the patterns which are defined in the configuration of the behaviors to be recognized. In order for a sequence to match, all patterns defined in the behavior must be recognized. Due to the earlier sensor fusion and classification of data, similar data items are combined thus removing similar but separate data items from the mining algorithm.

With sequence matching a degree of completeness is obtainable knowing how many terms make up a given pattern, the duration of each term, and the number of patterns. For instance in the 'At Work, Coding' behavior defined in figure 4, which is a single pattern composed of two terms. If the first term is matched, a fifty percent completion rating is assigned to the outcome of the behavior matching process. It is only when the entire duration is fulfilled on the second and final term that a complete match is found. Evaluating the progress of completion for each of the potential behaviors provides a means to estimate the predictability of one behavior over another. Two behaviors with almost identical patterns to recognize but one with a higher progress indicates the success of matching one behavior over another.

It is the state of a given behavior that is one of the primary solutions for the system. The current status or situational awareness for an individual at this point within the processing of the engine is trivial: the value at any time of each fused sensor data type is the situational state. The recognized behaviors are recorded for further analysis, integration with other systems, and interpretation by the user. The data mining process is continual as sensor data is input into the system and the process repeats.

Data Mining						
Type	**0500**	**0530**	**0600**	**0630**	**0700**	**0730**
Movement	Idle		Driving		Idle	
Application	Thesis Research					
Cursor	Active		Inactive			
Location	Home				Work	
Behavior Monitoring						
Writing Paper	Writing Paper		30%			
At Work, Coding	50%		0%		At Work, Coding	
Driving To Work	0%		Driving To Work		0%	

Figure 7. Behavior recognition example.

Presentation

The presentation or solution the engine develops is an updated representation of state determined from the ongoing collection of sensory data. This updated state is known as the situational or current state, which can be used in situational awareness based applications. Questions' regarding what a person is doing and what is occurring within the environment is the core of situational awareness. The engine is also responsible for pattern matching at which point it may match the entirety of a pattern of sensory data or only a portion thereof. The pattern matching is used for behavioral monitoring or commonly occurring activities performed by an individual.

Each solution created by the data mining process is stored back within the database for historical purposes. The history of data aids in providing additional solutions through the presentation layer. The presentation component of the system is used to generate or output the following: the current and historical status and behaviors monitored. The following illustration in figure 8 gives a detailed example of the classified sensor data and recognized behaviors over a period of a morning. The historical data shown below is obtainable by performing a query of the results over a period of time. The current situational and behavior state is available for any point in time.

<table>
<tr><th colspan="7">Historical Status</th></tr>
<tr><th>Sensor Type</th><th>0500</th><th>0530</th><th>0600</th><th>0630</th><th>0700</th><th>0730</th></tr>
<tr><td>Movement</td><td colspan="2">Idle</td><td colspan="2">Driving</td><td colspan="2">Idle</td></tr>
<tr><td>Application</td><td colspan="6">Thesis Research</td></tr>
<tr><td>Cursor</td><td colspan="2">Active</td><td colspan="4">Inactive</td></tr>
<tr><td>Location</td><td colspan="2">Home</td><td colspan="2"></td><td colspan="2">Work</td></tr>
<tr><th colspan="7">Historical Behavior</th></tr>
<tr><th>Behavior(s)</th><th>0500</th><th>0530</th><th>0600</th><th>0630</th><th>0700</th><th>0730</th></tr>
<tr><td>Behavior</td><td colspan="2">Writing Paper</td><td colspan="2">Driving To Work</td><td colspan="2">At Work, Coding</td></tr>
</table>

Figure 8. Historical status and behavior results.

Status Solution

The system defines two possible outputs which are generated by the data mining process and presentation layer. One of the outputs is the status or situational awareness component. The status is a collection of all fused sensor data types, their current value, and current evaluated duration. Using the example historical data from figure 8, the situational awareness result is shown in figure 9 for the time value of 0730. The state for a particular time depends on when the last change in value occurred and duration for that state. Situational state which changes frequently would therefore have more historical values but shorter durations for each momentary state.

Situational Awareness			
Sensor Type	**Time**	**Duration**	**State**
Movement	0700	0030	Idle
Application	0500	0230	Thesis Research
Cursor	0615	0145	Inactive
Location	0700	0030	Work

Figure 9. Situational awareness state for time 7:30 A.M.

Behavior Solution

The other output of the system is a collection of matched behaviors and their likelihood of completion. Figure 10 is the behavior monitoring result as given for a particular period of time. The example shows one behavior was matched completely, while another is only partially complete meaning the sequence of events defining that behavior has not fully occurred. Other behaviors which exist within the configured behaviors indicate a 0% status due to none of the potential patterns matching. The solution indicates completed behaviors, the time at which the behavior started, and the duration.

Behavior Monitoring			
Behavior	**Time**	**Duration**	**Status**
Writing Paper	-	-	30%
At Work, Coding	0700	0030	100%
Driving To Work	-	-	0%
Grocery Shopping	-	-	0%
Playing Sport	-	-	0%
At Work, Taking Break	-	-	30%

Figure 10. Behavior monitored state for time 7:30 A.M.

CHAPTER 4

SYSTEM GAPS

Security

Security issues surround any system involved with collecting personal data therefore government policies were created dictating the management of health data: the Health Insurance Portability and Accountability[28]. Like social websites, associating an anonymous identifier with collected personal data would be one of many precautions. Using a system to detect and automatically update routine behaviors to social sites has privacy issues due to the nature of the data being visible to others. The implementation of the system described in this paper makes no attempt to handle data securely nor does it address any of the issues surrounding collection and protection of private data. Covering information security and privacy is beyond the scope of current research regarding this system.

Personal Data Misuse

Overlooked by those involved or considering the application of self-monitoring services is the misuse of information. Balancing caloric intake from advice taken from software systems can easily be misused leading to more severe health problems [15]. As is the case with any system providing health or well-being feedback; the analysis and actions taken based on the data must be applied with caution. Current status that is available through the system for instance is a direct representation of data collecting from inputs into the system: therefore, erroneous sensor data will misrepresent the final situational status solution.

Sensor Data Requirements

Possibly an obvious impedance to appropriate system use is the requirement for sensory data. The non-obvious problems with sensor data requirements begin with have a variety or plurality of sensory devices and services. It is not a requirement to have lots of data, rather it is more important to have a reasonable variety for the configured system. Situational status and behavior recognition are driven by the data types available and rate at which data is available. The sampling frequency or rate sensor data is generated will affect how accurately results are depicted. The design of this system does not attempt to correct for missing data, instead the previous sensor state is carried forward into the next sensor state change detected. Status changes which are configured to be evaluated every minute would therefore likely miss a state change which occurred every 30 seconds (too fast for the system to detect). The implications of missing rapidly changing sensory data can be alleviated somewhat by mostly analyzing the results of macro state changes which do not occur as frequently.

Pattern Learning

In the data mining component of the system a method for using known patterns to detect reoccurring behaviors is described. The issue with using a fixed set of patterns is the inability to detect new behaviors or locate patterns which were not previously programmed into the system. For simplicity and producing a prototype system it was necessary to use a fixed set of patterns or behaviors to identify. However, with supervised learning techniques additional behaviors could be learned over time with interaction with the user [22]. The inability to detect non-configured behaviors is beyond the initial scope and is a limiting factor within the system design. By using various learning techniques the scale of the system could grow significantly resulting in a more robust system capable of providing additional status and behavior solutions.

CHAPTER 5

SYSTEM APPLICATIONS

Situational Awareness

Situational awareness is one of the two solutions which can be obtained from the system. It represents updated status regarding an individual at any point in time. The types of status and meaning that can be conveyed are determined by the types of sensory input or providers available. Blood pressure, heart rate, weight, and other biometrics would be useful in scenarios where health related information is useful to monitor: diabetic status monitoring, heart rate monitoring, and patient status. Temperature, heart rate, ambient light level, detected sounds would be useful in military scenarios or battlefield awareness applications. The status is a highly specialized process and some [10] have hypothesized could exist completely within the body as tiny processing modules. However, as given by this application it is provided by separate pieces of hardware and does not provide historical data unless specifically queried.

Raw sensor data is not available due to the sensor fusion and classification process only the fused sensor data is stored within the database which represents the current status. The fused sensor data is considered intelligent data due to the user-friendly nature of the values. However, no automatic interpretation of the data is performed. According to data fusion principles and practices [11] situational awareness should benefit from human and machine interaction rather than strictly automated. The human element and thus personal involvement in analyzing situational data is therefore dependent per individual [1]. For example, the environmental temperature of 65 degrees Fahrenheit may be hot for one individual but cool for another. This preference in analysis of data drives the design of the system to allow for a dynamic and configurable classification system.

Behavior Monitoring

Behavior monitoring is another solution provided by the engine. As sensory data is collected and repeated patterns within the data are found conclusions are built based on the configured behaviors. It is the behavior monitoring which provides insight into an individual's life. With behavior monitoring solutions, it is possible to infer what daily activities a person participates in, such that a user of the system can determine how an individual is spending their time. Health studies, behavior correction, project management, and other fields would all benefit from automatic tracking of an individual's behavior. Both short term and long term benefits related to health, personal fulfillment, goal reaching, knowledge, and other beneficial aspects can be obtained through self-monitoring technology.

Collecting information about how we live our lives and further analyzing it, can lead to uncovering patterns or interesting events [6]. A system for improving processes based on behaviors and how they relate to our environment is known as continuous improvement [7]. Continuous improvement is useful in reducing redundant activities and eliminating wasteful ones. For individuals interested in social networking applications behavioral data can be linked to accounts such that they are continually updated automatically with daily activity. As stated earlier within the design requirements automatic interpretation of behaviors is not available. The generated behavior predictions must be manually reviewed.

CHAPTER 6

CONCLUSION

System Test

For proper testing of the system design and evaluation of results many consumer products and a few custom products were used. In figure 11, the self-monitoring product classification was applied to known products for determining a diverse selection for testing. The system itself is implemented as a C# webserver with an in-memory database indexed by time. Sensory data input is pushed from external providers: website, mobile phone, desktop application, embedded sensory device, etc. Additionally data is pulled by internal tasks through the use of web services of remote sites.

A FitBit One [3] pedometer was used initially to track steps taken. Upon further review, it was no longer solely used for tracking movement but employed in conjunction with an accelerometer based mobile phone application. A Withings Wireless Weight [29] Scale was used for tracking measured weight. A mobile phone is used with GPS application provider component to push location data to the system site. A custom application was written for monitoring desktop activity relating to currently opened application and mouse activity. Another custom application uses the current situational status data for location to query the local weather conditions for the individual being monitored.

Additional custom devices would have been used if the hardware and software were readily available from similar activity tracking projects. The classification rules were set for individual preferences according to basic opinion polling, web research, and common fitness values. The remaining classification rules were left at default values. The behavior configuration was kept relatively minimal due to the availability of sufficient hardware. Larger scale tests with the system involving a multitude of device and service classes was therefore not exercised. The behavior configuration is modeled for a fictional user of the system with realistic behavior patterns defined.

Product Summary					
Product	Device Class	Service Class			
		Biometrics	Mood	Perception	Behavior
Fitbit One	Wearable	X			
Nike+ Sportwatch	Wearable	X			
Withings Blood Pressure Monitor	Portable	X			
Apple iPhone w/ Mood Tracker App	Portable		X		
Motorola Droid w/ Location Tracker App	Portable				X
Withings Wireless Scale	Placeable	X			
iFit WiFi Module	Placeable	X			
Biotronik Biomonitor	Implantable	X			
Flightscope X-Series	Placeable			X	
Oral-B Triump 9910	Portable				X
HAPILabs HAPIfork	Portable				X
HAPILabs HAPIbutton	Portable		X		
HAPILabls HAPItrack	Wearable	X			
iHealth Blood Pressure Monitor	Wearable	X			
Lark Sleep Monitor	Wearable				X
Adidas miCoach Heart Rate Monitor	Wearable	X			
iBGStar Blood Glucose Sensor	Wearable	X			
InteraXon Muse	Wearable		X		
Everspring Door / Window Sensor	Placeable			X	
Custom 1 'Weather' Sensor Provider	Placeable			X	
Custom 2 'Application' Sensor Provider	Placeable				X
Custom 3 'Thought' Tracker App	Portable		X		

Figure 11. Product summary with device and service classes.

Results

Custom hardware from similar projects relating to uncommon sensor data types and analysis would have been useful to test, but not critical. The design of the system is such that it allows new sensor types to be added easily. Sensor data was collected from the products described earlier with their respective sampling rates.

A few issues occurred initially when using a faster sampling rate to perform web service requests; the requests were denied due to excessive use of the system. The request rate was reduced and the accessibility problems were solved. Sensor data similarity measurements were tested with the collected data and worked as expected. Sensor data values which were considered similar in nature were discarded and replaced with a fused sensor data element indicating start time, duration, and value. A heading change of only a single degree for instance over a minute timeframe did not meet consideration for a new sensor data value; instead, the value was combined leaving the original heading for a longer duration.

Classification rules are currently hard-coded within the classification process but implemented in a manner that allows extension. Due to time constraints within the software development lifecycle of the system prototype the data mining component had to be manually tested. The simplicity of the data mining process allows for easy evaluation and prediction of behaviors based on test data (included within this document). Since fused sensor data is stored within chronological order and sequential data mining is performed by sensor type the process is almost trivial for relatively small data sets.

The situational awareness state variables were successfully evaluated according to system specification. The situational awareness state as defined earlier is the classified sensor data, upon verification of the classification algorithm working the validity of the current status is a given, they are the same. Testing of the predictive behaviors algorithm involved manually evaluating the sequences of classified sensor data. Because the format of the behavior patterns are effectively regular expression matching sequences of data the system output was simple to ascertain and verify. Similarly defined patterns were tested to ensure behaviors were tested with known matching, and partial matching outcomes.

An unexpected result of the predictive behavior solution was discovered when applying the data mining process. One case involves behaviors which are sparsely defined typically indicating patterns are not complete for intended results. A theoretical scenario of this case has several devices: a blood pressure monitor, pill dispenser monitor, and stress level monitor. A behavior is defined to track sequences of the pill dispenser indicating use or the blood pressure monitor reading favorable results. In one instance the pill dispenser is not used, stress level indicator is, and the blood pressure monitor reads favorable results. This instance indicates the stress level indicator is missing from the definition of the patterns regarding this behavior.

Another case involves behaviors which are exhaustively defined. A scenario of this case has two devices: a weight scale and a fitness tracking device. A weight monitoring behavior is defined to indicate weight loss when sampled weight goes down or weight gain when sampled weight goes up over a period of time. The weight monitoring behavior has been occurring resulting in weight loss detected. The fitness tracking device does not report a status change from its previous state indicating no activity. In this specific example an emergent behavior is discovered: the fitness tracking device has not been taken with the individual when they went to exercise (resulting in weight loss, but not activation of fitness tracker). No pre-programmed behavior was defined as indicating forgetting fitness tracking device. A simple, but powerful observation is made through this example. If a behavior learning algorithm was implemented it is possible the undefined behavior would have been readily apparent and added to the available configuration dataset.

Lastly, more consumer products would have been used if sufficient funds were available. During the research conducted for this paper various manufacturers' product launches were tracked to determine when new self-monitoring products would be available. At least 4 products were made available during the research period: a smart watch with pedometer, a food consumption rate tracking fork, a mood tracking push-button, and a physical activity indicator wristband. The product availability could indicate a trend in technology growth rate benefiting the system capabilities and allowing further expansion.

Future Work

The applications of self-monitoring are growing as quickly as technology improves to adapt new sensor capabilities into smaller components. Using new types and more reliable sensory technology improves further the software which it is based upon. Some of the devices and services used in the design of the system are provided for reference within figure 11. By selecting an assortment of self-monitoring tools, more types of sensory data is available to the system.

By creating a general process for collecting, interpreting, storing, analyzing, and projecting results allows for a robust system to be created. Additionally, sensor data types may be added; new behavior defined to recognize additional patterns; which result in new behaviors to understand. Improving ones daily life or routine through self-monitoring technologies involves a process of monitoring, studying, and acting upon the data to get results [27].

However, behaviors are more than workplace routines or dieting trends; they are the way in which humans live their lives. The applications of knowing an individual's current state as it relates to their environment [9] could aid in health-monitoring, social networking applications, military applications, directed marketing, and many other uses. Healthy living [9] and behavioral detection and correction [2] are applications for behavior monitoring and situational awareness statistics. Beyond technological solutions exists general answers to the questions: where we have been, what we are doing, and where we are going.

REFERENCES

[1] Blasch, E. Kadar, I. Salerno, J. Kokar, M.. Das, S. Powell, G. Corkill, D. Ruspini, E. (2006) "Issues and Challenges in Situation Assessment (Level 2 Fusion)". Journal of Advances in Information Fusion. Volume 1, Issue 2. December 2006.

[2] Buettner M., Philipose M., Prasad R., Wetherall D. (2009) "Recognizing Daily Activities with RFID-Based Sensors". Proceedings of the 11th international conference on Ubiquitous computing, New York, New York, USA, ACM Press, 2009, 51-60.

[3] Fitbit (2012) Fitbit App Gallery, Internet:. http://www.fitbit.com/apps

[4] Google (2011) Google Health. Internet: http://www.google.com/intl/en_us/health/about/index.html

[5] Hand, D., Mannila, H., Smyth, P. (2001) "Principles of Data Mining". MIT Press. Cambridge MA. August 2001.

[6] Han, J., Kamber, M., Pei, J. (2011) "Data Mining: Concepts and Techniques". Morgan Kaufmann; 3rd Edition. July 6, 2011.

[7] Imai, M. (2012) Gemba Kaizen: "A Commonsense Approach to a Continuous Improvement Strategy". McGraw-Hill Professional; 2nd edition, May 23, 2012.

[8] JDL (1991) "Data Fusion Lexicon". Data Fusion Sub-panel of the Joint Directors of the Laboratories." U.S. Department of Defense 1991.

[9] Krejcarek, B. (2011) "Sticking To It: Using Sensors to Help Us Reach Everyday Goals" (TEDx), Silicon Valley, California.

[10] Kurzweil, R. (2012) "How to Create a Mind: The Secret of Human Thought Revealed". Viking Adult, November 13, 2012.

[11] Liggins, M. Hall, D. (2009) "Handbook of Multisensor Data Fusion; Theory and Practice", 2nd Edition. CRC Press; 2nd edition. 2009.

[12] MariaDB (2013) Internet: https://mariadb.org/.

[13] Microsoft (2012) Microsoft Health Vault.Internet: http://www.microsoft.com/en-us/healthvault/

[14] Microsoft (2013) Microsoft SQL Server.Internet:. http://www.microsoft.com/en-us/sqlserver/default.aspx

[15] Moore, C. L. (2001) “The Dangers of Self-Monitored Dieting: What are our patients really doing?” Cleveland Clinic Journal of Medicine. September 2001, Volume 68, Number 9, 781.

[16] Machulis, K. (2011) OpenYou Libraries. Internet: http://www.openyou.org/libs/

[17] Najafi, B., Aminian, K., Paraschiv-Ionescu, A., Loew, F., Bula, C.J., Robert, P. (2003) “Ambulatory system for human motion analysis using a kinematic sensor: monitoring of daily physical activity in the elderly”. Biomedical Engineering, IEEE Transactions. Volume 50, Issue 6.

[18] Newman, A., Mitzel G. (2013). “Upstream Data Fusion: History, Technical Overview, and Applications to Critical Challenges”. Johns Hopkins APL Technical Digest. Volume 31, Number 3. 2013.

[19] Parkka, J. Ermes, M. Korpipaa, P. Mantyjarvi, J. Peltola, J. Korhonen, I. (2006) Activity Classification “Using Realistic Data From Wearable Sensors”. Information Technology in Biomedicine, IEEE Transactions. Volume 10, Issue 1.

[20] Quantified Self (2012) Quantified Self: Knowledge Through Numbers. Internet: http://quantifiedself.com/about/

[21] Rosenberg, B. (2009) “Harnessing the full power of sensor fusion”. Defense Systems, Knowledge Technologies and Net-Enabled Warfare. September 15, 2009.

[22] Russel, S., Norvig, P. (2010) “Artificial Intelligence: A Modern Approach”; 3rd edition. Pearson Education, Inc. Upper Saddle River, NJ.

[23] Klein, L., (2012) “Sensor and Data Fusion: A Tool for Information Assessment and Decision Making”; 2nd edition. SPIE Press Monograph PM222.

[24] Spriggs, E. Frade, F. Hebert, M. (2009) “Temporal Segmentation and Activity Classification from First-person Sensing”. IEEE Workshop on Egocentric Vision, CVPR 2009, June 2009.

[25] SWI-Prolog (2013). Internet: http://www.swi-prolog.org/

[26] Thompson, K. Regular Expression Search Algorithm. (1968) Communications of the ACM. Volume 2, Number 6, 419.

[27] Trickler, C. (2013) “An Overview of Self-Monitoring Systems”. SAIS 2013 Proceedings. Paper 37.

[28] U.S. Dept. of Health & Human Services (2003). “Summary of the HIPAA Privacy Rule”. Internet: www.hhs.gov

[29] Withings (2012) Withings. Internet: http://www.withings.com/en/scales.

[30] Wolf, G. (2010) “The Quantified Self”, Technology Entertainment Design (TED), Cannes, France.

APPENDIX

Software Prototype

The following are screen captures of the user interface designed for the software implementation of the system. The application was developed in C# on the .NET platform version 4.5

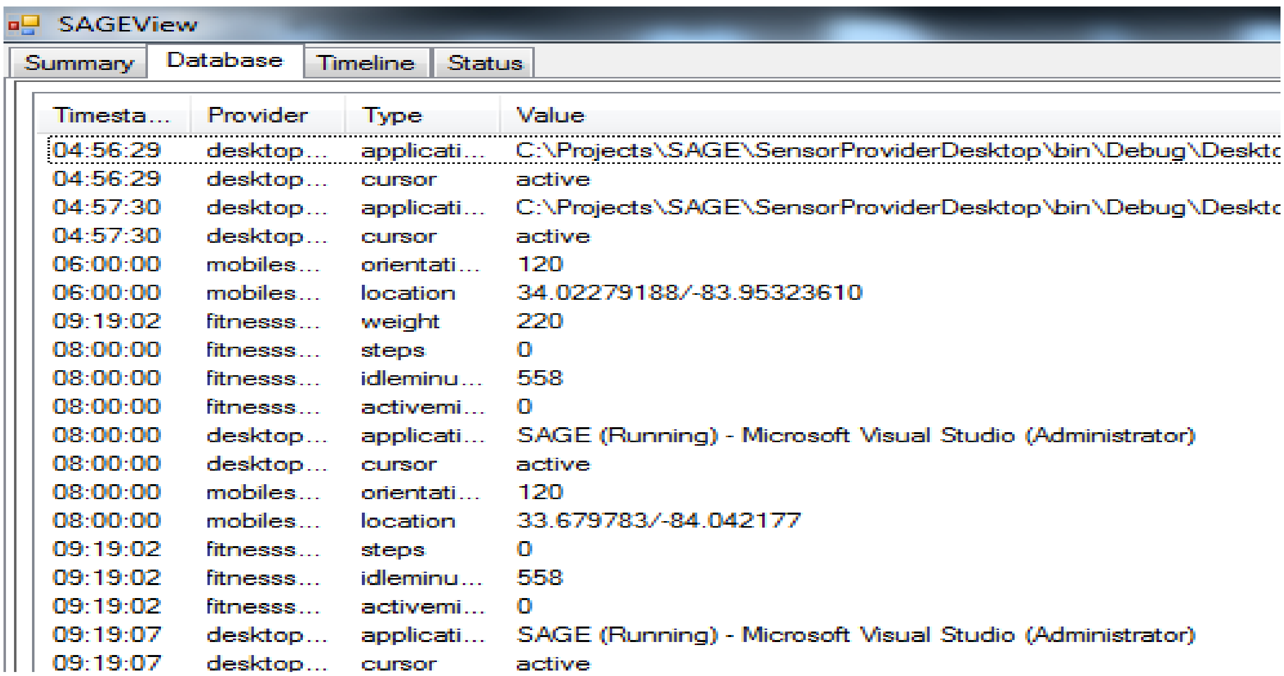


Figure A1. View showing historical raw sensor data.

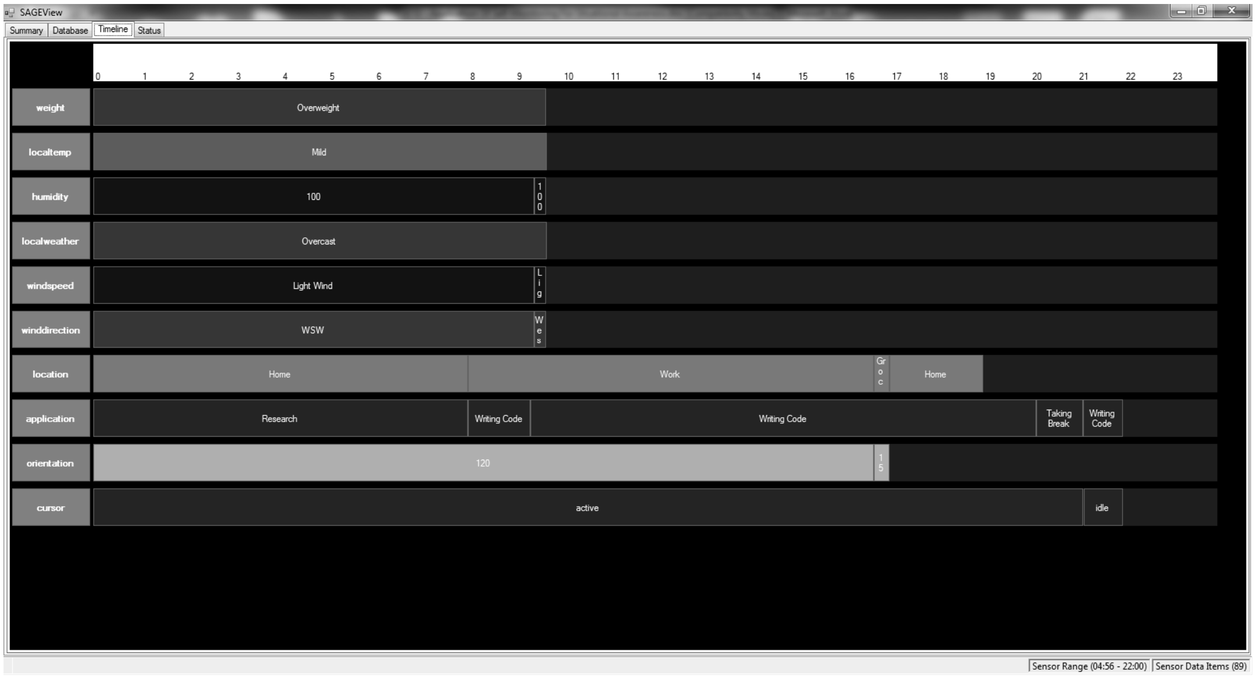


Figure A2. View showing classification of data for a given day.

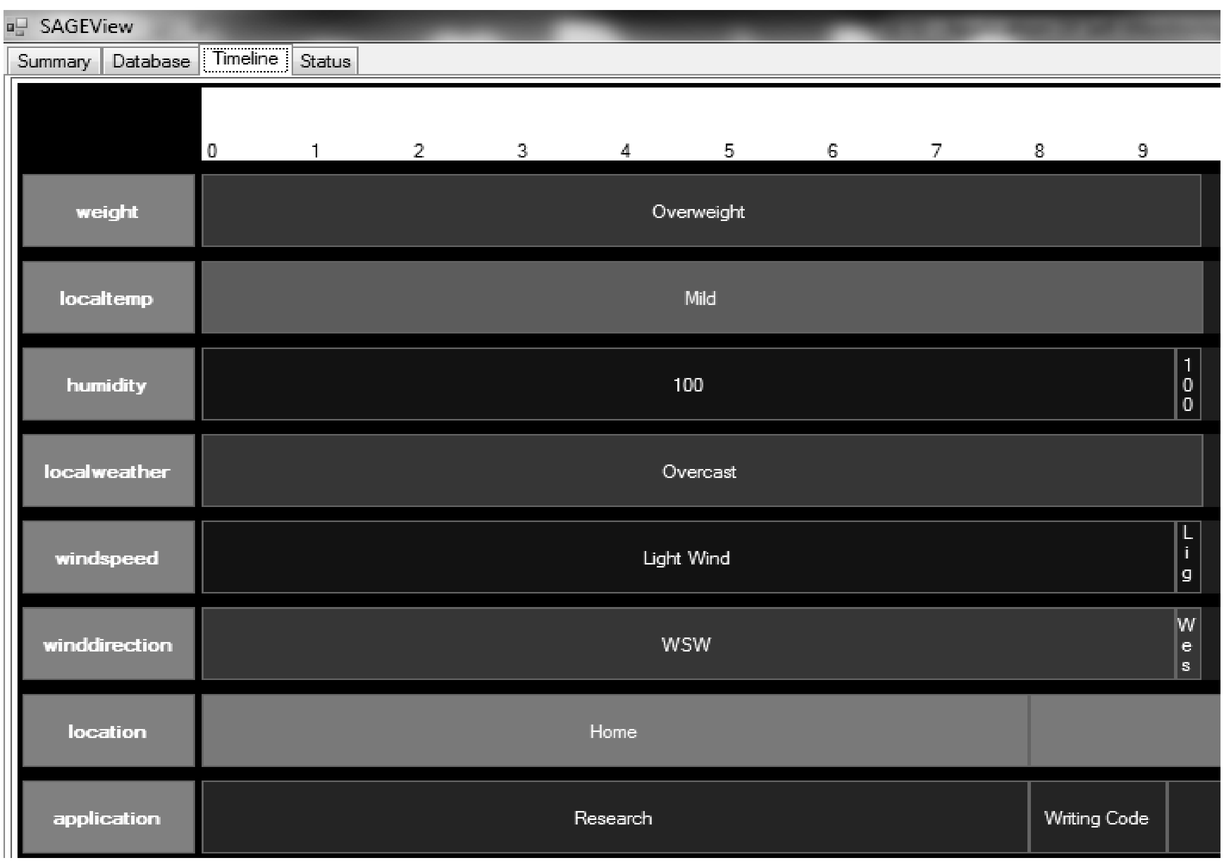


Figure A3. View showing details of classified data for a given day.

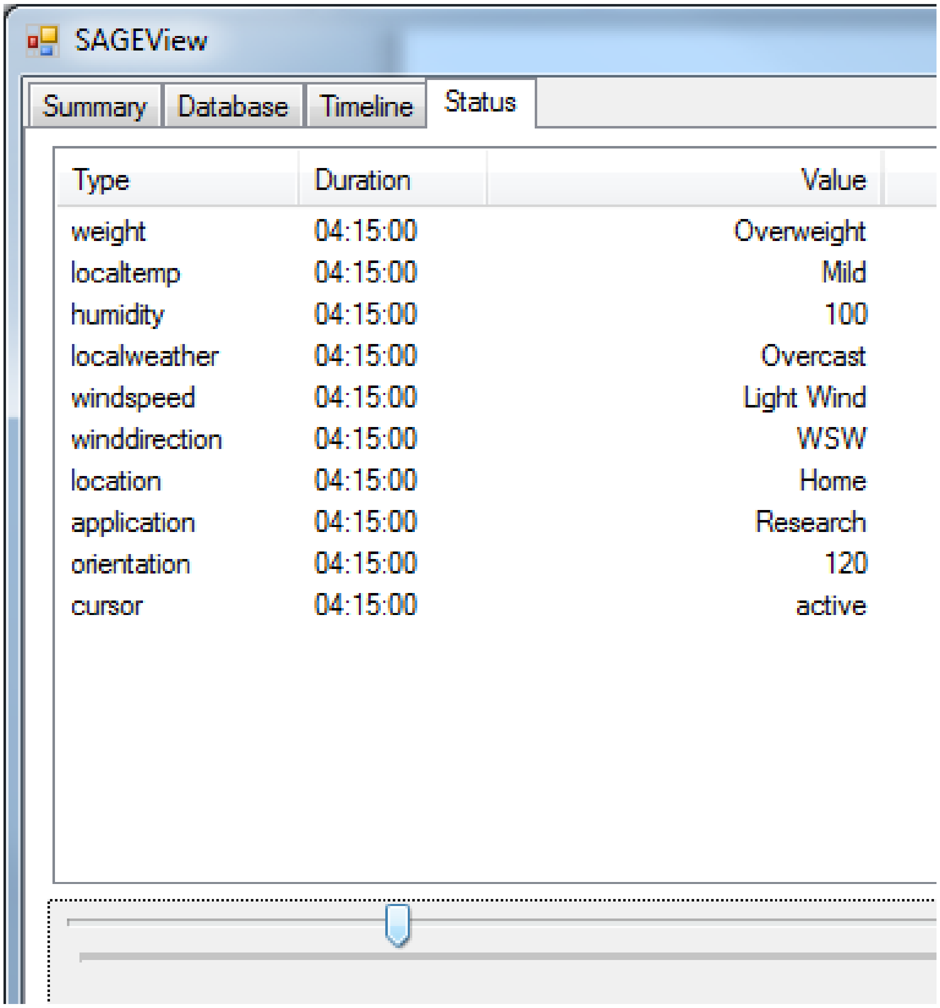


Figure A4. View showing situational status for a given time.

```
private void mobilePush()
{
        // function is periodically invoked on mobile phone

        // create web objects
        HttpClient httpclient = new DefaultHttpClient();
        HttpPost httppost = new HttpPost("http://thinkoutcloud.com");

        try
        {
           // add sensor data to push operation
           List<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>(2);
           nameValuePairs.add(new BasicNameValuePair("client", "mobilesensor"));
           nameValuePairs.add(new BasicNameValuePair("location", String.format("%.8f/%.8f",
                                                             location_lat, location_lng)));
           nameValuePairs.add(new BasicNameValuePair("orientation", String.format("%.0f", orientation)));
           nameValuePairs.add(new BasicNameValuePair("lux", String.format("%.2f", light)));
           nameValuePairs.add(new BasicNameValuePair("proximity", String.format("%.2f", proximity)));
           httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));

           // execute http post
           httpclient.execute(httppost);
        }
        catch (Exception e)
        {
        }
}

@Override
public void onLocationChanged(Location location)
{
    // capture and log location changes through gps sensor provider
     try
    {
            // update current location
            location_lat = location.getLatitude();
            location_lng = location.getLongitude();
     }
     catch (Exception e)
    {
    }
}

public void onSensorChanged(SensorEvent event)
{
        // collect each type of additional sensor data for push operation
         switch(event.sensor.getType())
         {
          case Sensor.TYPE_ORIENTATION:
                 orientation = event.values[0];
          }
}
```

Figure A5. Code listing for mobile sensor data provider.

```
private static void fitnessPull()
{
    // invoke fitbit api client request
    var fitbit = new FitbitClient(consumerKey, consumerSecret,
                                  credentials.AuthToken, credentials.AuthTokenSecret);

    var profile = fitbit.GetUserProfile();

    // convert from kg to lbs
    weight = (int)(profile.Weight * 2.20462);

    var activity = fitbit.GetDayActivity(DateTime.Now);

    // collect useful sensory data from account
    steps = activity.Summary.Steps;
    idlemins = activity.Summary.SedentaryMinutes;
    activemins = activity.Summary.LightlyActiveMinutes +
                 activity.Summary.VeryActiveMinutes +
                 activity.Summary.FairlyActiveMinutes;


     // now that pull request has completed from fitness website, push to engine

     // execute http post
     using (WebClient client = new WebClient())
    {
        // add sensor data to push operation
        byte[] response = client.UploadValues("http://127.0.0.1:8080/", new NameValueCollection()
        {
               { "client", String.Format("{0}", "fitnesssensor") },
               { "weight", String.Format("{0}", weight) },
               { "steps", String.Format("{0}", steps) },
               { "idleminutes", String.Format("{0}", idlemins) },
               { "activeminutes", String.Format("{0}", activemins) },
          });
      }
}
```

Figure A6. Code listing for fitness sensor data provider.

```
public override SAFusedData Classify(SAFusedData dataitem)
{
        return new SAFusedData(dataitem.Timestamp,
                               dataitem.Duration, dataitem.SensorType,
                               dataitem.SensorValue);
}
```

Figure A7. Code listing for default data classification.

```
public override SAFusedData Classify(SAFusedData dataitem)
{
        // scan through each classification range
        foreach (KeyValuePair<string, SAClassRange> kvp in _dataclasses)
        {
                // if fused sensor data is within this range
                // classify it as new sensor type and assign label
                double sensorvalue = Convert.ToDouble(dataitem.SensorValue);
                if(
                sensorvalue >= kvp.Value.min &&
                sensorvalue <= kvp.Value.max
                )
                {
                        return new SAFusedData(dataitem.Timestamp,
                        dataitem.Duration,
                        dataitem.SensorType,
                        kvp.Key);
                }

        }

        return null;
}
```

Figure A8. Code listing for range based data classification.

```
// compare current 'fused' data item with new data item to be added
fused = _fuseddata[dataitem.SensorType][_fuseddata[dataitem.SensorType].Count - 1];
current = new SAData(fused.Timestamp, dataitem.SensorProvider, fused.SensorType,
fused.SensorValue);

// perform sensor data type specific comparison
if (SAEngine.Significant(current, dataitem, ref duration))
{
	// if there is a significant change - add to fused list

	// update previous item duration
	fused.Duration += (dataitem.Timestamp - (fused.Timestamp + fused.Duration));

	// create the new fused data item
	fused = new SAFusedData(dataitem.Timestamp,
				new TimeSpan(0, 0, 0),
				dataitem.SensorType,
				dataitem.SensorValue);

	  // significant change, new fused data item
	_fuseddata[fused.SensorType].Add(fused);
}
else
{
	// if it is the same item - update existing timespan
	fused.Duration += (dataitem.Timestamp - (fused.Timestamp + fused.Duration));
}
```

Figure A9. Code listing for data fusion.