DATA SOURCES FOR A MODEL TO PREDICT AIR TEMPERATURES INSIDE A RESIDENTIAL HOME

N Bausch

Systems Engineering Research Group, School of Engineering, University of Portsmouth, PO1 3DJ.

Email: nils.bausch@port.ac.uk

Abstract

This paper introduces the data sources for a model to predict air temperatures inside a residential home. Sources included local and remote environmental sensors. Data from the sensors is being used to create models of temperature change. The sensor data is expected to lead to new and novel system designs that will combine the new models with traditional heating control to create a new optimal start-stop heating application for residential homes.

Keywords: domestic, Smart Home, prediction, AI, sensor fusion, smart environment, thermal comfort.

Introduction

Data collected for this research used various sensor types and differing underlying technologies. This Paper introduces hardware, software, data retrieval methods and selected sensors that comprised the system of disparate sources. The Paper first introduces hardware sensors that were installed in a residential home and software that was used to collect the data. The Paper continues with environmental sensor data retrieved from websites. The Paper concludes with the data retrieval and storage of the sensor data.

Local sensors

Sensors were installed in a residential home in locations such as within and on walls, under the roof, outside, and interfaced two electronic systems: an alarm system and heating control. The sensors and the methods of data access are presented in this Section. The location of the residential home can be seen on a map shown in Figure 1.

Wire system. The 1-wire system was a communication bus-system developed by Dallas Semiconductor. The 1-wire bus connected devices with two wires: one wire for ground and one wire for data and voltage supply. This was possible due to a built-in capacitor in each device that provided power during data communication. The 1-wire sensor devices in this research work used three wires with a dedicated power supply and a master node to access the network. The use of three wires was necessary due to the number of sensors in the network and the network length to ensure timely data communications. 1-wire sensor devices were generally inexpensive and included for example: digital thermometers, switch sensors, and counters. Each device had a 64 long unique identification number, which was used to address devices individually. The following three paragraphs introduce device families that were installed in the residential home in addition to a master device to control the network.

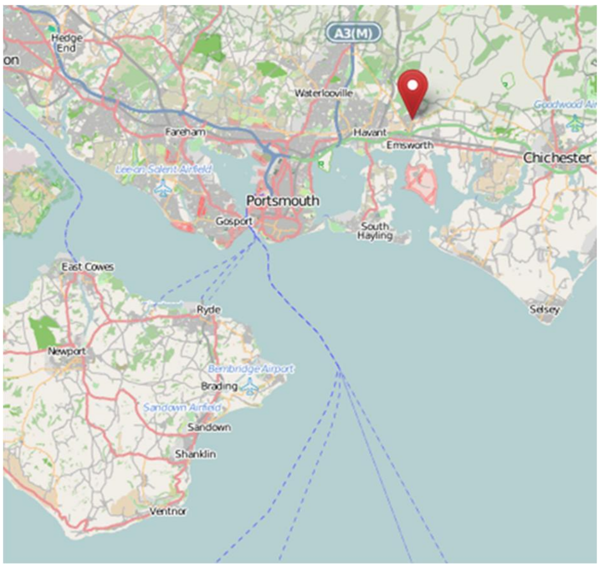

Figure 1: Location of the residential home in the south of England, UK in Emsworth: 2013

Digital thermometers. The residential home had devices installed prior to the start of the research work in 2009, which were used to collect temperature readings from a swimming pool. These devices were digital thermometers from the device family DS18B20.

The digital thermometers had a resolution of up to 12, which enabled a measurement resolution of 0.0625°C. The TO-92 with its small size enabled an easy deployment of the sensors in walls, plastic casings and outside the residential home. Sensor installations can be seen in Figure 2, showing a bare sensor and an outside installation at the residential home.

Figure 2: Temperature sensor DS18B20

Switch sensors: The DS2408 (see Figure 3) was a switch sensor with eight channels. The channels were user configurable as inputs or outputs. A different two channel switch sensor in use was from the DS2506 family. The switch sensors were used in interfaces to third party electronic devices.

HA7net. The HA7net was an Ethernet to 1-wire interface, providing a web interface to access 1-wire devices. The HA7net provided HTML websites to retrieve and send data to the bus system. Functions that were implemented on the HA7net and applied in this research work were as follows:

- Search: probed the connected 1-wire network and returned all devices or a specific family of devices as a list of unique IDs.
- Write: sent commands to the 1-wire network and returned replies from devices where applicable.
- Temperature read: was a higher level function, which bundled the retrieval of temperature data of single or several devices into a single function.

The HA7net was able to support an overall network length of 300 with up to 100 connected devices. A picture of the HA7net can be seen in Figure 4.

A diagram showing the HA7net and a 1-wire network can be seen in Figure 5.

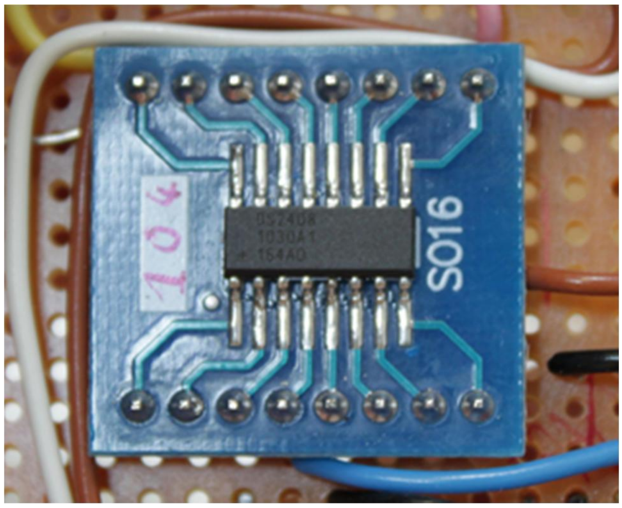

Figure 3: DS2408 on an adapter board

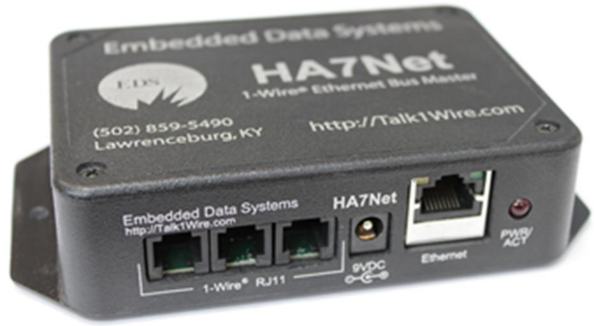

Figure 4: HA7net with three RJ-11 ports for 1-wire and one RJ-45 port for Ethernet

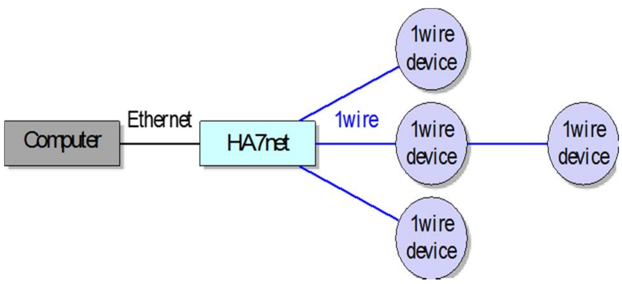

Figure 5: Example network layout of HA7net with connected computer and 1wire devices

On-site weather station.

Davis Instruments produced a range of weather stations that collected weather data either via wireless or wired communication. The weather station installed at the residential home was a wireless Vantage Pro 2. The weather information was transferred every minute to a console inside the residential home and sent to a website via an Ethernet module. An additional software package enabled the download of historical data from the website. See Figure 6 for a weather station and various sensors, such as anemometer, rain collector, wireless transmitter, and console.

Figure 6: Davis Vantage Pro 2 weather station with anemometer, solar panel and wireless transmitter, radiation shield with air temperature sensor, rain collector, and indoor console davisnet: 2013

Remote sensors

Remote sensors were defined as sensors in the locality of the residential home, which provided environmental data from third party sources. The remote sensors were accessed via their websites, which are described in the following Subsections.

Weather forecast. Websites, such as the Met Office, the BBC and Yahoo provided access to localised weather forecasts as RSS or as a web page. The weather information on these web pages was retrieved through an API or through matching text from the website to extract data.

The websites provided different environmental weather information, various forecast lengths, and meteorological details. For example the BBC weather forecast website provided daily forecasts for one, two, and three days. The Met Office website provided daily forecasts of up to five days and hourly forecasts for up to 24 hours.

Weather station. The website for the on-site weather station featured a map and access to the weather station if this was permitted by the weather station owner.

Other local sources of weather conditions were buoys and weather stations collecting data for a nearby harbour in Chichester, Hampshire UK. The buoy websites made available environmental data similar to land based weather stations, in addition to sea conditions.

Software background

Data collection was carried out with different operating systems, programs and programming languages. Their purpose in the process of data retrieval and their interactions are described here.

Operating systems. Various operating systems were used to access and process the collected sensor data. This was due to the applied programming languages and the setup that was in place before the research work commenced. A Microsoft Windows Server at the residential home collected data from the on-site 1-wire network. The server was accessible from the internet to permit access to a SQL database downloads and visualisations of the recorded data. A Microsoft Windows XP virtual machine was used to import the downloaded SQL database. The websites' data was recorded on a Linux Gentoo server at first, and a secondary system running FreeBSD was set up later on. The final steps of analysis and modelling were carried out on a Mac OS X system in Matlab.

Programming languages & programs. On-site data collection was achieved with a VB.NET program running on the Microsoft Windows Server, named TempReader, which saved the data to a SQL database. The program accessed the website of the HA7net device on the LAN. A preliminary version of this program was available prior to the start of the research in 2009. This version was enhanced by including libraries from the HA7net to implement high level functions. The program was then completely rewritten to support an oop design and went through several iterations of improvements and enhancements. An ASP.NET program running on the Microsoft Windows Server was used to create and download backups of the SQL database. An additional ASP.NET program residing on the on-site server, called Grapher, created plots and tables with statistics from the collected data. The programming language Ruby was used on a Linux and FreeBSD servers to access the websites, namely weather stations and forecasts. The scripts collected and extracted textual data found on these websites and saved them in separate files for each website.

System implementation

In this Section the disparate sources of information used in the research work are introduced, consisting of the 1-wire devices, the website forecasts, and the website weather stations. A graph detailing the data retrieval process can be seen in Figure 7. The diagram also shows the data import into the software package Matlab, which was used for further data analysis.

1-wire devices. A DS2408 switch device was used to interface to a Galaxy 8 house alarm system. The following signals of the house alarm system were interfaced: sounding of the exit horn, alarm unset, alarm set, and intruder status.

A second DS2408 switch device was used in a comfort input box. The comfort input box was used to input the inhabitants' personal thermal comfort level. The supported thermal comfort levels were: cold, ok, and hot. The comfort input box recognized up to five individuals.

A third DS2408 switch sensor was used as an interface to the heating system from Heatmiser. Thermostats in the house signalled a heating demand to the Heatmiser system. By placing a DS2408 onto the Heatmiser control board, the sensor was able to record heating demand. A second sensor (DS2406) was on the same board.

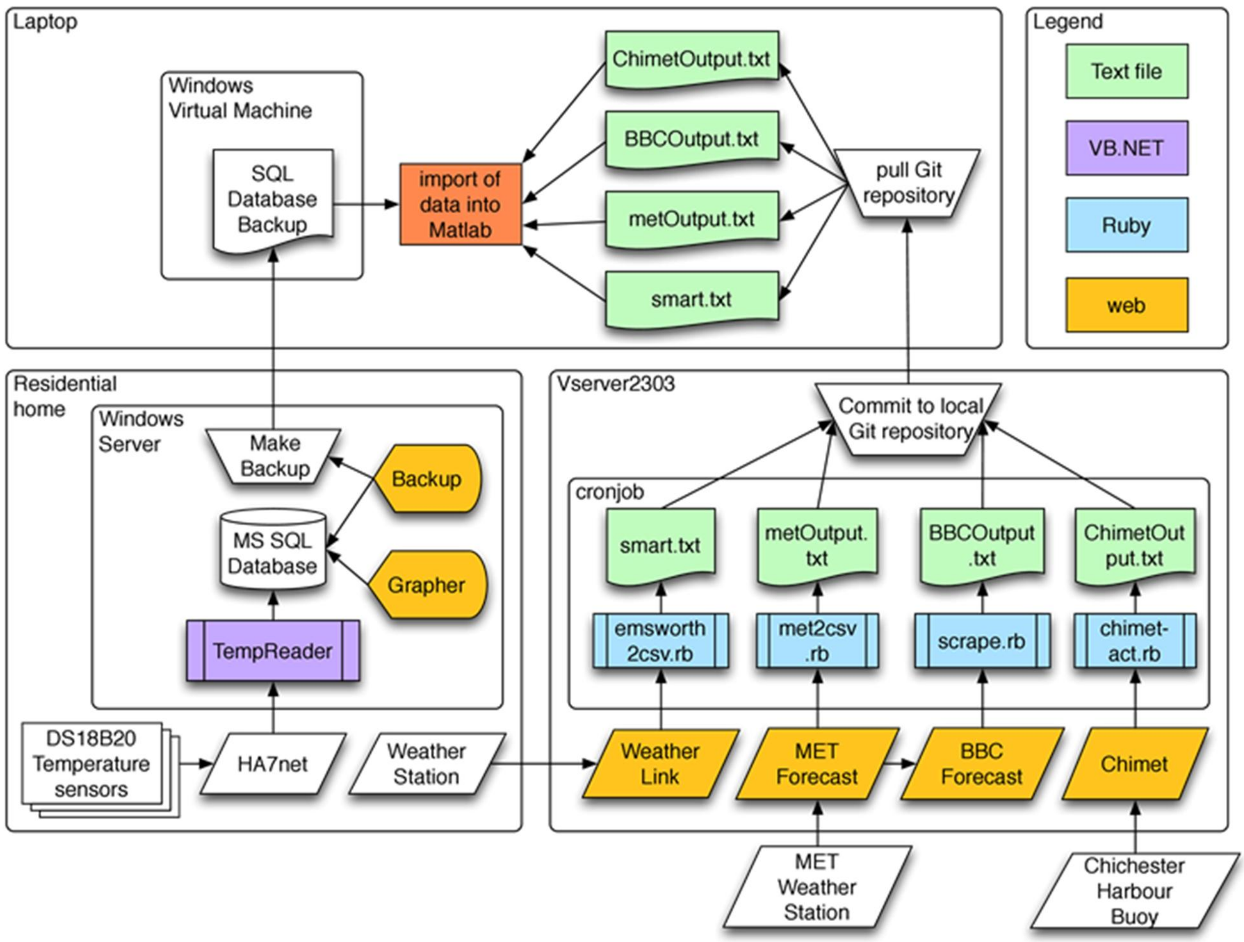

Figure 7: System overview of data sources, showing the sensor data locations in the bottom part and on the top part the data assembly and import into Matlab

The second sensor was intended to control the Heatmiser heating system by signalling a heating demand, but was not used in this research work due to time constraints. Several DS18B20 temperature sensors were installed inside and outside the residential home. The inside installation comprised of sensors embedded into walls, sensors inside enclosures to collect air temperatures, and sensors exposed to ambient temperatures in different parts of the residential home. The outside temperature sensors were embedded into walls and inside of enclosures to minimize heat radiation affecting the ambient temperature measurements.

On-site weather station. A weather station from Davis Instruments was installed above the apex of the residential home in autumn 2010 to collect local meteorological data. The data was transmitted to and displayed on a console inside the residential home. The console submitted the data every minute to a website which made it accessible on:

http://www.weatherlink.com/user/smart. The website made available historical data with an interval of 30 minutes, therefore the website with current data was web scraping to collect data in shorter intervals.

The WeatherLink website can be seen in Figure 8.

Table 1: Weather station sensors

Measurement name
wind speed
wind direction
rain collector
solar radiation
air temperature
relative humidity
barometric pressure

Weather forecasts

Meteorological forecast data from the BBC and the Met Office were used. The websites provided up to date information on weather forecasts in the area of the residential home. The BBC website provided a three day forecast, including the current day. Measurements taken from the website are shown in Table 2.

Table 2: BBC forecast website weather information

Measurement name
wind speed / direction
air temperature
relative humidity
barometric pressure
sunrise/sunset date and time

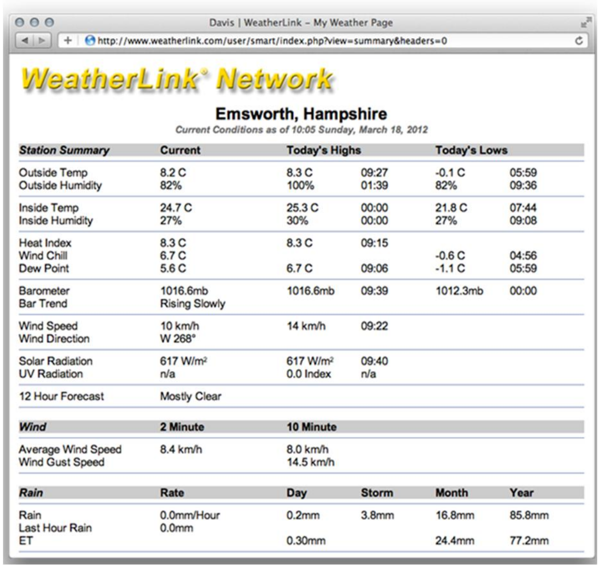

Figure 8: WeatherLink website screenshot showing meteorological data transmitted from the on-site weather station

The Met Office provided a five day forecast, including the current day. The forecasts were updated between two to four times an hour. The weather station site for the Met Office measurements was located approximately 21 miles away from the residential home. Measurements that were taken from the website are shown in Table 3. An example screenshot of the actual website can be seen in Figure 9.

Table 3: Met Office forecast website weather information

Measurement name	Unit
wind speed	miles per hour
wind direction	directions as letters
air temperature	celsius
visibility words	as scale

Current weather. In addition to the on-site weather station, data from third party weather stations was collected. The first one was installed on a buoy and the second one was a weather station in Portsmouth. The latter was only used to create and test a prototype web scraper for the on-site weather station.

Chimet. The website http://www.chimet.co.uk hosted weather information collected 1.61 Km offshore, close to Chichester harbour entrance. The buoy was approximately 11 Km away from the residential home.

Table 4: Chimet buoy weather information

Measurement name	Unit
air temperature	°C
barometric pressure	-

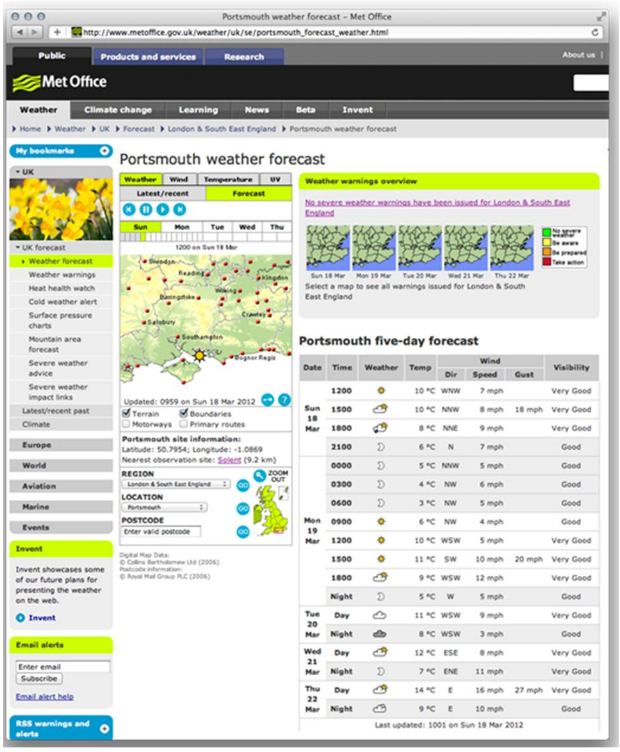

Figure 9: Met Office five day weather forecast website

Weather station on Portsmouth island. A weather station was accessed via its Weatherlink website.

The website was used to develop web scraping scripts that were later used for the on-site weather station.

The weather station's website frequently stopped updating for days and therefore was not suitable for data collection.

Retrieval of sensor data

The sensors were categorised by their technologies and differed in their locations in relation to the residential home.

The 1-wire network was locally installed and the sensor network data stored on-site.

The current weather conditions and forecasts were saved and retrieved on computers located at remote locations.

This Section describes the implementation of the data retrieval and storage of the sensor data.

1-wire sensors. Sensors used during this research are listed in Table 5, which shows the frequency of measurement and the date the sensors became active.

The manufacturer of the HA7net created a VB.NET class library to access the HA7net's websites and common 1-wire device families.

During the development and integration of new sensors, the VB.NET program underwent several changes and additions.

Table 5: On-site 1-wire sensors with the device family, read interval, and date of first recorded data

Location	Type	Reading every
East Wall Outer Sensor	DS18B20	30
East Wall Inner Sensor	DS18B20	30
First Floor Core Temp	DS18B20	30
Silver Tube Air Temp	DS18B20	15
First Floor Landing Air	DS18B20	15
Ground East Inner wall	DS18B20	30
Ground Air	DS18B20	15
Boiler Cold Water	DS18B20	5
Boiler Hot Water	DS18B20	5
Boiler CH Flow	DS18B20	5
Boiler CH Return	DS18B20	5
Eaves N-E Air	DS18B20	30
Roof Under Tiles N-E	DS18B20	30
Roof Under Tiles N-Mid	DS18B20	30
Eaves N-Mid Air	DS18B20	30
Wall N-Mid Room Side	DS18B20	30
Wall N-Mid eaves side	DS18B20	30
Comfort User 1 Hall	DS2408	10
Comfort User 2 Hall	DS2408	10
Comfort User 3 Hall	DS2408	10
Comfort User 4 Hall	DS2408	10
Comfort User 5 Hall	DS2408	10
Alarm Status Sensor	DS2408	5
Heating Zones	DS2406	5
Roof Air	DS18B20	30
Roof Apex	DS18B20	30
Ground East inner Wall2	DS18B20	30
Ground Air2	DS18B20	15

The program was able to carry out the following tasks:

1. Retrieve all sensors from the database and initialize them.
2. Check for network connectivity of the HA7net and check each sensors' availability.
3. Initialize a scheduler with the lowest read interval.
4. Scheduled reading of sensors through the HA7net.
5. Saving of the values in a SQL database and logging of errors.
6. Reset schedule timer for next invocation. Repeat from 4.

The program communicated with an SQL server and retrieved initialisation data from the Sensors table. This table contained following fields: a unique ID used as a reference for connected tables, the unique ROM ID of the sensor, an unused type field, a RefreshRate as sensor read interval in minutes used by the scheduler, an unused accuracy value, and a location note as a description.

Data itself was saved in the TempData table and used the unique ID from the Sensors table, a timestamp, and a value. The data was saved by a stored procedure, which was a program written in T-SQL and saved inside the SQL database. The stored procedure achieved data compression, by comparing the previous two data values against each other and against the latest value. Depending on timestamps and the values, following outcomes were possible:

1. If the previous two data values had different values, then the latest value would be inserted into the database.
2. If the previous two data values had the same value and were not recorded within one hour then the latest value would be inserted into the database.
3. If the previous two data values had the same value and were recorded within the last hour, the latest value's timestamp would be used to update the previously last value's timestamp. Only in this case the data was compressed, by updating existing data values' timestamp.

An example of this compression, comparing both original collected data and saved data at the end of the process. The application of compression created data records with irregular read intervals. The original data with regular intervals was recreated with the refresh time during the data import. The time of one hour was chosen to ensure the detection of errors. The highest data compressions were achieved with sensors being read every five minutes, reaching 23%. Sensors with a read interval of 15 minutes reached up to 17% and sensors with a read interval of 30 minutes achieved a compression of 5%.

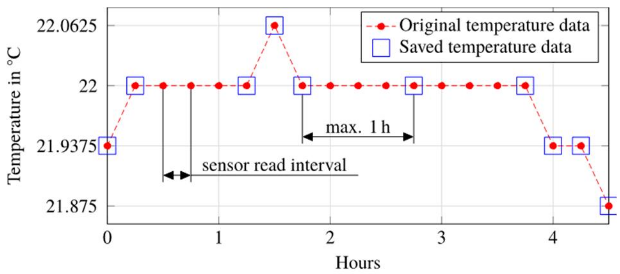

Figure 10: Example data compression showing original and saved data

Errors from the sensors were logged in an Errors table and included the unique ID from Sensors, a timestamp, a string error message and an ErrorID.

The SQL server data backup was accessed with an ASP.NET web-interface from:

http://phd.nilsbausch.net/Backup,

which was written by a research group member.

In autumn 2011 a collaborative development with a research group member was started. The output was a graphing tool for historical 1-wire sensor data with a web interface and downloadable graphs in PNG format. The aim was to create a tool, which made available the sensor data and error logs from a remote location. The Grapher tool displayed error logs, latest temperatures, and minimum and maximum values of sensors. It was also possible to plot diagrams of different sensors, time spans, and customizations for the output diagrams. A graph can be seen in Figure 11.

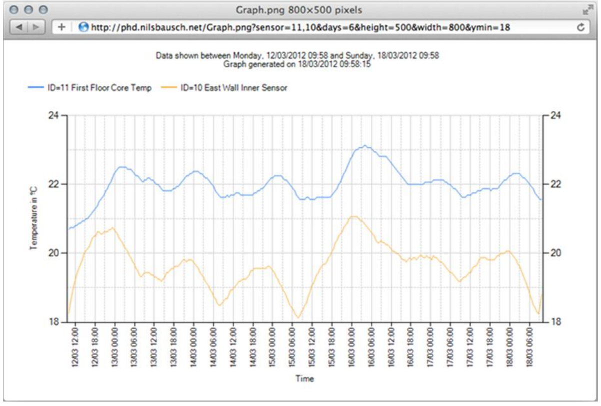

Figure 11: Grapher demo showing PNG output generated by passing arguments to the URL query in the URL bar on top

Weather forecasts and weather station

The websites were web scraped with regexp to extract the data, for example temperatures, wind speeds, wind direction or pressure. The web scraping generally progressed in three steps:

1. Access the website via its URL.
2. Retrieve data of interest with regexp.
3. Save the retrieved data and a timestamp in a CSV file. If the website provided forecasts, a forecast date was recorded as well.

The web scraping scripts were written in Ruby. The scripts were invoked at set intervals to assure that updates on the websites would be detected and saved. The intervals were set as follows:

1. Every minute for the weather station website. The weather station sent updates to the WeatherLink website every minute.
2. Every four minutes for the Chimet buoy website. The website stated, that the data was updated every five minutes.
3. Every five minutes for the Met Office update. The Met Office website was updated between three to five times an hour.
4. Every full hour for the BBC website. The BBC weather feed was updated three to four times a day.

Each of the websites was accessed, the data retrieved, and recorded in separate files.

Each file had a headline which listed the data types and units.
A headline with example data is shown as follows:

```
publication_date,forecast_date,temp_C,win
d_direction,wind_speed_miles
2010-07-21T15:01:00+01:00,2010-07-
21T16:00:00+01:00,20,SW,16
```

The filenames for each website were:

Met Office forecast metOutput.txt
BBC forecast BBCOutput.txt
Chimet buoy ChimetOutput.txt
Weather station smart.txt
Davis weather station davis.txt

As the files were saved in a remote location, a backup was made every few weeks with Git to make the data accessible in Matlab for further data manipulation.

Vacant space and heating sensing

The research described in this Dissertation concentrated on vacant space heating. The model criteria ‘vacant space’ was introduced in April 2011. The concept behind vacant space was to decrease the dynamics when collecting temperature readings inside the residential home. Presence detection of the inhabitants through an alarm system enabled vacancy detection. Sensors integrated into a boiler collected information on the heating system, enabling heating detection. Both sensors together enabled the detection of vacant space and the heating state of the residential home and were an integral part of the rules set out for defining datasets for the creation of prediction models.

Conclusion

This Paper introduced the data collection methods used in this research. Data was accessed and recorded from disparate sources in local and remote locations.

The residential home had sensors installed at the beginning of the research and several additional sensors were installed to collect data of interest, such as temperatures of brickwork, interfacing the alarm system, and inside air temperature. Other data sources were third parties, such as weather station and forecast data, which were accessed through websites. An on-site weather station was used to collect climate data at the residential home. This diverse set of sensors used different technologies and interfaces. Each specific interface to the data sources were presented and described in this Paper.

The sensors defining vacant space and heating states were introduced in April 2011, marking the start date for data used in later analysis. The hypothesis was, that diverse sources of information would contribute to the creation of a model which was able to produce usable forecasts.

Next steps will include processing of the collected data, creating of models with viable methods, and benchmarking of the new models against known data to measure their forecast abilities.