Internet of Smart Things
A Study on Embedding Agents and Information in a Device

Leo van Moergestel[1], Melvin van den Berg[1], Marco Knol[1], Rick van der Paauw[1], Kasper van Voorst[1], Erik Puik[1], Daniël Telgen[1] and John-Jules Meyer[2]
[1]*Institute of ICT, HU University of Applied Sciences, Utrecht, Netherlands*
[2]*Information and Computing Sciences, Utrecht University, Utrecht, Netherlands*

Keywords: Agent-based IoT, Device Information, Life Cycle Dependent Information.

Abstract: The term Internet of Things (IoT) is used for situations where one or more devices are connected to a network or possibly the Internet. Most studies focus on the possibilities that arise when a device is capable to share its data with other devices or humans. In this study, the focus is on the device itself and what kind of possibilities an Internet connection gives to the device and its owner or user. Also the data the device needs to participate in a smart way in the IoT are part of this study. Agent technology is the enabling technology for the ideas introduced here. A proof of concept is given, where some concepts proposed in the paper are put into practice.

1 INTRODUCTION

Nowadays, many devices are connected to a network, be it the Internet or another implementation of network connectivity. The connectivity may be based on wireless technology like Bluetooth or WiFi, but wired solutions are also possible. The connectivity is used by a device to share or exchange data with other devices or humans. This so-called Internet of Things (IoT) (Ashton, 2009) is considered to be the next step in the direction of a world where everything is connected. Most devices are offering data or accepting data to operate in a certain environment. In this paper the focus will be on using agent technology as a basis for devices operating in the IoT.

The rest of this paper is organised as follows: Section 2 is dedicated to IoT. Several classifications and implementations are presented as well as the added value that is offered by IoT. In Section 3, after a short introduction of agent technology, the classes of IoT devices where agent technology can be applied are discussed. The reason why agent technology fits in the IoT model is also explained in Section 3. The roles of agents and the data a device needs to operate in the IoT depending on the phase of the life cycle are the subjects of Section 4. The use phase of the life cycle of the product will be the phase that will have the main focus, but the roles of the agents in other phases are mentioned as well. In this section the technical possibilities are also discussed.

The implementation is the subject of section 5. Here the device to be produced is introduced and the parts it consists of. This will lead to a device description format that is both computer- as well as human-readable. This device description will be enhanced by assembling instructions that will be the basis of the information about the manufacturing. Next the implementation of some ideas proposed in Section 4 will be explained. Section 6 discusses related work. A conclusion and a bibliography will end the paper.

2 IoT

This section will discuss IoT. First, classifications of IoT devices are presented followed by a short motivation what the merits are by connecting devices to the Internet.

2.1 Classifications of IoT Devices

Three types of classification will be presented here:

1. classification based on administration;
2. classification based on connectivity;
3. classification based on the soft- and hardware capabilities of the device.

A possible way to classify IoT devices is looking at their administration. By this is meant the way the devices know each other and what they are expected to

Moergestel, L., Berg, M., Knol, M., Paauw, R., Voorst, K., Puik, E., Telgen, D. and Meyer, J-J.
Internet of Smart Things - A Study on Embedding Agents and Information in a Device.
DOI: 10.5220/0005659801020109
In *Proceedings of the 8th International Conference on Agents and Artificial Intelligence (ICAART 2016) - Volume 1*, pages 102-109
ISBN: 978-989-758-172-4

do in the IoT environment. Five different classes of IoT systems are presented here.

1. Fully static systems. All the devices in the system are pre-configured to communicate and interact with each other. An example is an industrial sensor network like WirelessHART (Song et al., 2008).
2. Centrally administrated systems with a single administration unit. In this case the devices are pre-configured to interact with a central unit - This is the registration step. Each device can query the central unit to identify other devices that it needs to interact with. Devices can communicate with each other directly or they can communicate via the central unit. An example of this class of IoT devices are devices in smart homes with capabilities for coordinating power usage, adapting to environmental conditions, ensuring security of the home, etcetera.
3. Centrally administered systems with multiple administrative units where the administrative units are centrally coordinated. These systems are typically widely distributed, hence several administrative units are used at different geographical locations. The devices are pre-configured to locate/interact with nearby administrative units. The administrative units keep track of all the devices and ensure that each device is aware of the other devices that it may need to interact with. An example is an infrastructure system for automated guided vehicles (AGV) in a wide geographical area.
4. Centrally administered systems with multiple administrative units where the administrative units form a decentralized system. The administrative units are mobile systems. These systems cooperate with each other to identify and track all the devices. The devices are made aware by the nearest administrative unit of the other devices they may need to interact with. Devices can communicate with each other directly or they can communicate via the administrative units. Devices themselves are typically mobile units. An example for this class of IoT are systems that keep track of children and old people to ensure their safety.
5. Fully autonomous decentralized systems. In this case each device is an independent unit that has capabilities to identify other units and coordinate with them. Devices communicate with each other directly. Devices can be fixed or mobile units - The autonomy is used to minimize the interaction time while ensuring requisite coordination and cooperation with other units. An example in this case are avionic systems: each aircraft can autonomously identify nearby aircrafts and coordinate its motion in order to prevent accidents.

Another way to classify IoT devices is to look how they are connected to the Internet, resulting in three different classes.

1. A device can be part of a special network, wired or wireless that is connected by a special network device that will be the gateway to the Internet. This situation occurs in the so-called sensor networks, where simple devices use a dedicated network technology to connect with each other or the gateway. Examples of such networks are WirelessHART (Song et al., 2008) and ZigBee (Ian Akyildiz et al., 2002).
2. A device can have its own Internet address but is only capable to reply to requests received from the network to send data.
3. A device can have its own Internet address and being capable to communicate directly and actively with other devices on the Internet.

The third classification that will presented here is based on the computing capabilities of the device itself. Two classes are important for our discussion.

1. A device has only a single thread of execution to make it work as an IoT device. Normally there is not much storage capability in this type of device. An example is a sensor or actuator that is connected to a network.
2. A device is capable to execute multiple threads, resulting in a system that has more than one process running. This type of device can even be capable to run an operating system and use some local storage.

2.2 Benefits of IoT

By connecting devices to the Internet, the following two possibilities arise:

1. Data collecting: data can be collected and analysed giving insight in all kind of situations where the IoT devices are involved.
2. Control: because a device is connected, it can be controlled. This results is three possibilities:
   (a) Communication: this is an extension of data collecting because the kind of data can be selected.
   (b) Interaction: remote control, updates and adjustments are possible.
   (c) Automation: a device can learn, but not in a stand-alone situation, but in the situation where

resources and other devices on the network can support the learning system.

To see where agent technology fits in the aforementioned classifications we will give a short introduction to agent technology and a definition of what an agent is as used in this paper.

3 AGENT TECHNOLOGY

This section will start with a short introduction to agents and the way they can be embedded in a device. The section will continue with a motivation for using agent technology. Finally, the types of IoT devices as introduced in the classification, where the agent model can be used are discussed.

3.1 Embedding Agents

An agent is an autonomous software system that is designed to play its role and to achieve one or more of its goals. Agents are autonomous entities that can be embedded in a device under the condition that the device itself contains a processor and memory to store the agent code as well as to execute this code. In practice this means that the device should be capable to run a kind of operating system. Two important solutions for embedding an agent in a device exist:

1. The agents can be a part of the device software infrastructure. Without the agent the device cannot operate properly. This situation applies when the agent is developed in combination with the device software and the device infrastructure itself is agent-based.
2. The agent can be a software system that operates separately from the device infrastructure itself. In this situation, the device will operate properly without the agent. An advantage is that the agent can be added at any time and the agent can even run in cyberspace. In the latter case a connectivity with the device is compulsory.

3.2 Why Agent Technology?

The main reason for using agent technology is the fact that an agent can operate without human intervention. It is autonomous in a sense. Intelligence can be a property of an agent as well as mobility. A mobile agent can move from one workspace to another opening the possibility to embed it in a product when the product is made, let the agent move to cyberspace when needed and let the agent go to another device. Agents can communicate, thus exchange information So if a mobile agent is not possible in certain situations, a newly created agent can take over valuable information from another agent.

3.3 IoT Device Types

As stated already, the device should have the capabilities to run the agent. This means that it must be capable to run multiple threads or even better processes. The device should be connected directly to the Internet, because when the device is hidden behind a gateway, as in the situation of some implementations of sensor networks, standard software techniques cannot be used to communicate with the agent embedded in the device. It depends on the situation if the device operates actively or just waits for requests. In regard to the administration, agents can be used in all five types mentioned. However, in practice fully static systems (the first type in the list) require human intervention in configuring. This approach conflicts with the idea that the system should work and configure itself autonomously.

4 AGENT ROLES IN IoT

In this section the focus will be on generic roles that an agent can play in a device. The roles depend on the phase in the life cycle of a product. First these phases will be introduced including the possible roles for a product agent. Next, for every role in every phase, the data or knowledge of the device is summarized. A more thorough discussion of agents roles in the life cycle of a product can be found in (Moergestel et al., 2010).

4.1 Phases in the Life Cycle of a Product

A device starts its life during manufacturing. Moergestel et al. (Moergestel et al., 2011) describes a production system where a product starts its life as a software agent that is responsible for the manufacturing of a single possibly unique product. This so-called product agent contains the blueprint of the product to be manufactured and is the basis for the concept of the IoT in other phases of the life cycle of the product it is embedded in. In Figure 1 the life cycle of an arbitrary product is shown. After the design, the product is produced in the manufacturing phase. After manufacturing, the product is distributed. Next comes a very important phase, being the use of the product and finally the phase where the product should be recycled. In all of these phases, the product agent can play a role that will be globally described in the next subsections. For every phase and associated role, an overview of information acquired or used is given. A more extended discussion can be found in (Moergestel et al., 2014). We remark that in the literature the term *product life cycle* (PLC) is commonly used to denote a concept that is different from what we call the *life cycle of a product*. The product life cycle can be defined as the process wherein a product is introduced to a market, grows in popularity, and is then removed as demand drops gradually to zero (Lilien et al., 2003).

Figure 1: Life cycle of a product.

4.1.1 Design and Manufacturing

A trend in modern manufacturing is that the design of a product will be greatly influenced by the individual end-user requirements. Nowadays cost-effective small scale manufacturing will become more and more important. This trend is enabled by Internet technology and new developments in manufacturing like 3D printing opening possibilities to adapt a manufacturing system to individual end user requirements. In (Moergestel et al., 2015) an implementation of manufacturing as a service (MaaS) is described. In this MaaS system the product agent is created. Important is the fact that the product agent is responsible for the manufacturing of the product as well as for collecting relevant production information of this product. This concept is the basis for the roles of the product agent in later phases of the life cycle. The product agent carries the product design as well as the production data and can be viewed as the software entity that represents the product in cyberspace. The information available should open the possibility for a the device to clone itself.

4.1.2 Distribution

In (Burmeister et al., 1997) a logistic application based on a multiagent system is described. Agent-based logistic applications have transport from source to destination as the main goal. However, other important data can be information about product handling and external conditions, like temperature, shocks etcetera. These data can be measured by cheap possibly wireless sensors and collected by the product agent in its role of guidance agent during the transport or after arrival at the destination. The handling and external conditions during transport can be important for the other coming cycles in the life phase, because they influence product quality thus having impact on maintenance and repair. The information available gives an overview in time of the physical and conditional data. The main problem related to this phase is powering the device during transport. That is the reason that in most cases it is better to collect the transport information at arrival, when de device is powered on. Low power battery-operated sensors already exist that can collect and store information during transport. These devices will collect the needed information and can be read by the product agent when it powered on.

4.1.3 Use

In the use phase, the product agent can perform actions that will help and support the end-user by keeping valuable information at hand, offering a user manual or preventing or minimizing the effects of misuse. By giving feedback about the usage and problems that occur during usage it can also help the manufacturer to improve its products. Finally in the recycling phase, the information collected during usage can be very helpful.

In the next paragraphs several topics of usage of the product agent are discussed.

Collecting Information. A product agent will collect information about the use of the product as well as the use of the subparts of the product. Testing the health of the product and its subsystems or subparts can also be done by the agent. These actions should be transparent for the end-user. To perform these actions, the product agent should be connected to sensors and software subsystems in the product.

Advising and Control. The information collected by the product agent gives a clue about the product use and the use of the subsystems. An agent can suggest maintenance or replacement of parts. The agent can advice a user about the usage of a product, this can be done directly but also automatically without direct user intervention. An example of the latter is: If a product needs resources like electric power, an agent can suggest a product to wait for operation until the cost of electric power is low i.e. during the night. It depends of course on the type of device if this should be implemented.

Maintenance and Repair. An agent can identify a broken or malfunctioning part or subsystem. This could be achieved by continuous monitoring, monitoring at certain intervals or a power-on self test (POST). Condition monitoring is the process of controlling the functioning of parts of a device or of the device itself. It is an important possibility to prevent malfunctioning. Three types of information play an important role: the manufacturing information, the transport information and the usage information. This is exactly the information that is collected by the product agent. Repairing a product is easier if information about its construction is available. Issues during transport have an influence on the quality as well as the way a product has been used.

Other Possibilities. Some possibilities that could be achieved by using a product agent are:

- Transparency of the status of a product after maintenance by a third party. The agent can report to the end-user what happened during repair so there is a possibility to check claimed repairs. Of course the agent should be isolated from the system during repair to prevent tampering with it.
- Recovery and tracing in case of theft or loss are also possible by using the embedded product agent technique.
- Advising the end-user who wants to replace a certain device by a new one. The product agent can give advice about the properties the replacing device should have, based on what the product agent has learned during the use phase.

4.1.4 Recycling

In Moergestel et al. (Moergestel et al., 2014) the role of the product agent in the reuse of subsystems and components is mentioned as well as in locating expensive or rare material.

Another advantage of having a product agent at hand in case of recycling is the fact that the product agent has the information how a product is constructed. This is helpful when a product must be taken apart. For certain steps a kind of undo-steps should be carried out to dismantle a product. These undo-steps, if they exist, can be incorporated at the moment the product is assembled.

4.2 Embedded Knowledge for Different Phases

The whole design including the user preferences that are taken into account form the basis of the knowledge or information of the product agent. The product agent will use this knowledge to guide the product to be made during manufacturing. This concept is treated in (Moergestel et al., 2010). The design will lead to sequences of production steps. In the manufacturing phase itself other knowledge is added to the knowledge base, especially the information about how the manufacturing has been carried out for all the production steps and what parameters where used. In the distribution phase, all kind of conditions met during transport can be collected. The transport route itself could also be of interest and can be enabled by the use of a global position system (GPS). This might be helpful to trace a device to its end-user. However, as mentioned before the device should be capable to operate during transport. In the use phase, user preferences during use and usage of subparts are interesting data to be collected. In the recycling phase, data about the whole manufacturing process is available. The information about the condition of parts or components is also known. This will help to identify parts that might be available for further use. In this case, the device should be operating. If that is not possible, the component where the agent is stored should be reachable so the information is available. Making backups in cyberspace during the use phase can be a solution for situations where the embedded agent is completely lost.

5 IMPLEMENTATION

To illustrate the concepts mentioned in the previous sections a device has been chosen that is capable to play Internet audio streams. For the end-user it will look like a radio device also know as an Internet radio. The goal of the work presented here is to embed a product agent together with the knowledge collected during manufacturing. The product agent will play its role in the use phase. It will collect information, advise the user and display information that can be helpful in case of repair or maintenance.

5.1 The Internet Radio

The Internet radio adheres to the requirements given in section 3. It is connected to the Internet, it has computing power available for extra processes and it consists of different components that can be monitored during use phase, making it a good test case to implement ideas mentioned in Section 4. The end-user can also give his or her preferences in the design phase. In Figure 2, the prototype of the Internet radio is shown. A block schematic is given in Figure 3. The device consists of a BeagleBone computerboard, a display, an audio subsystem connected to an USB-port, speakers and a casing. The main electronic parts are shown in Figure 4.

Figure 2: Radio.

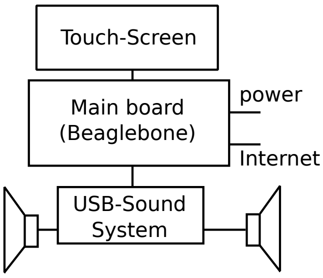

Figure 3: Radio.

Figure 4: Main electronic parts of the radio.

5.2 XML Description

All information is stored in XML-format. The main reason for using XML is that it is a widely accepted open standard. A lot of libraries for many programming environments exist. An alternative could have been JSON. Some examples of information storage will be presented in this section. In the design phase a top-down composition of the radio is given. The radio consists of six panels (front, back, top, bottom, left and right) with zero or more attachments. The end-user can select a shape, resulting in these six panels. This results in a toplevel XML-file where the radio is defined to consist of a these six panels in combination with actions describing how to assemble the radio using these panels.

```
<Radio>
  <Source>
    six panels
  </Source>
  <Actions>
    assembly instructions
  </Actions>
</Radio>
```

Details of the panels themselves should be added. A small abbreviated part of the XML-file describing the front panel of the radio looks like:

```
<Front>
  <Source>
   <Panel>
     info about panel
   </Panel>
   <Attachments>
    <Speakers>
      info about position and type
    </Speakers>
    <Display>
      info about display position
    </Display>
   </Attachments>
  </Source>
  <Actions>
    info about construction
  </Actions>
</Front>
```

The panel info part contains the file used by the water jet cutter device that was used to construct the panel and also the material used. An instruction video or instruction list can be part of the information in the Actions-section. The components that are attached to the panel are also included in the Attachments-section. Again a video instruction or instruction list in the Actions-section can help to construct the complete part.

What is now available in this file is information about the parts, the way they are made, the way they should be combined, in fact the production steps. After the manufacturing phase an extended XML-file is created containing information when it was actually constructed including the production feedback. The manufacturing phase consists of a sequence of production steps. Every step is carried out by production equipment or by human workers. The results of the steps are collected by the product agent in the extended XML-file. In the last step the product agent will be embedded in the product together with all the information collected so far. At this stage is is also possible to add information about reversing the production steps if possible at all. This information can be helpful to disassemble the device at the end of its life cycle or for repair or maintenance.

There are two types of user-specified properties that will be added to the knowledge base of the product agent being the XML-file. The properties that the user selected when the design of the radio was made and the properties the user specifies during its use. The first type of properties are mostly fixed if it concerns hardware properties, while the second type may change during use. The shape and colour of the case is fixed, but settings and preferences controlled by software can change. An example of the user preferences made during design that might alter during usage, is the layout of the control display. In Figure 5 an example is given how the display arrangement can be made by the end-user using a web-interface and generating an XML-layout file to be used by the GUI software. This information is also included in the XML-file.

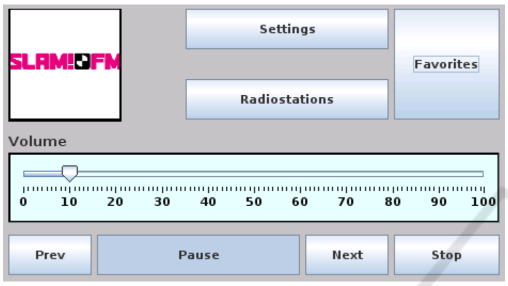

Figure 5: Display and touchscreen.

5.3 Embedding the Product Agent

In the prototype an existing agent platform has been used. The platform Jade (Bordini et al., 2005) was chosen for the following reasons:

- Jade is Java-based. Java is a versatile, widely supported and powerful programming language;
- Because Jade is Java-based it also has a low learning curve for Java programmers;
- Jade offers possibilities for agents to negotiate. If we need extra capabilities, the Jade platform can easily be upgraded to an environment that is especially designed for BDI agents like 2APL (Dastani, 2008) or Jadex (Bordini et al., 2006). Both 2APL as well as Jadex are based on Jade but have a more steep learning curve for Java developers;
- The Jade runtime environment implements message-based communication between agents running on different platforms connected by a network.
- In Jade, agents can migrate, terminate or new agents can appear. This feature is important for the implementation described here.

When information is embedded, the question arises how much detail should the information give. In the example of the radio, the electronic heart of the system is a BeagleBone single board computer. This computer will be considered as a building block without details about its components like chips, capacitors and other electronic parts. This could be included in a separate product agent belonging to the board itself. If such an agent exists a method of contacting it is the only thing needed. If there is no such agent available, the information should be accessible by using Internet technologies in another way. This method can be used for mass-produced components and prevents the situation where all this information is replicated.

So far, the radio contains the design and manufacturing information as well as the information collected during usage. The list of preferences is adjusted to the usage of the device by the end-user. The information is available to the Internet and can be inspected using a web-interface.

6 RELATED WORK

This work started by investigating new manufacturing methods as described in (Moergestel et al., 2011). The agent-based manufacturing opened the method to preserve the manufacturing information per product. Agent-based manufacturing has been described and developed by Bussmann. In the work of Bussmann a work-piece agent resembles the product agent proposed in our paper. An important difference is that the work-piece agent ceases to exist when the product has been made. Among other work that focusses on agent-based manufacturing, the work of Paolucci should be mentioned here (Paolucci and Sacile, 2005), because this work gives a good overview of the possibilities that arise by using agent technology. The possibilities of using the agents in the whole life cycle of the product are described in (Moergestel et al., 2010). In (Moergestel et al., 2013) a system is presented that embeds a product agent in an already complete product that supports repair and reuse of material.

An example of energy saving using multiagent technology is described by Ruta e.a. (Ruta et al., 2012). This work has the focus on domotics and agents play a role in negotiating to lower the energy usage in a domotic environment. In this case the agents are not part of the devices themselves as described in our paper, but they represent energy consuming devices.

7 CONCLUSION

This paper described the possibilities of Internet-connected devices, when agent technology is added to make the device a powerful and also versatile component in the IoT. Embedding and collecting information enables possibilities that will benefit the end-user as well as the manufacturer and the environment. It makes the task of data mining more reachable because the agent can select interesting data and it helps to improve products. Storing data can be done in cyberspace as well as in the device itself depending on the circumstances like internal storage capabilities and continuous network connection. In the prototype presented in this paper only a few possibilities are implemented. In future research, other possibilities mentioned in this paper should be implemented.

REFERENCES

Ashton, K. (2009). That 'the internet of things' thing. *RFID Journal*, (22 july).

Bordini, N., Dastani, M., Dix, J., and Seghrouchni, A. E. F. (2005). *Multi-Agent Programming*. Springer.

Bordini, R., Braubach, L., Dastani, M., Seghrouchni, A., and Gomez-Sanz, J. (2006). *A survey of programming languages and platforms for multi-agent systems*. The Free Library.

Burmeister, B., Haddadi, A., and Matylis, G. (1997). Application of multi-agent systems in traffic and transportation. *IEEE Proceedings on Software Engineering 144 (1)*, pages 51-60.

Dastani, M. (2008). 2apl: a practical agent programming language. *Autonomous Agents and Multi-Agent Systems*, 16(3):214–248.

Ian Akyildiz, I., Su, W., Sankarasubramanian, Y., and Cayirci, E. (2002). A survey on sensor network. *IEEE Communications Magazines, August*.

Lilien, G., Kotler, P., and Sridhar Moorthy, K. (2003). *Marketing models*. Prentice-Hall of India.

Moergestel, L. v., Meyer, J. J. C., Puik, E., and Telgen, D. (2010). The role of agents in the lifecycle of a product. *CMD 2010 proceedings*, pages 28–32.

Moergestel, L. v., Meyer, J. J. C., Puik, E., and Telgen, D. (2011). Decentralized autonomous-agent-based infrastructure for agile multiparallel manufacturing. *ISADS 2011 proceedings*, pages 281–288.

Moergestel, L. v., Meyer, J. J. C., Puik, E., and Telgen, D. (2013). Embedded autonomous agents in products supporting repair and recycling. *Proceedings of the International Symposium on Autonomous Distributed Systems (ISADS 2013) Mexico City*, pages 67–74.

Moergestel, L. v., Meyer, J. J. C., Puik, E., and Telgen, D. (2015). Implementing manufacturing as a service: A pull-driven agent-based manufacturing grid. *Proceedings of the International Conference on ICT in Education, Research and Industrial Applications (ICTERI 2015) Lviv Ukraine*, pages 172–187.

Moergestel, L. v., Puik, E., Telgen, D., Folmer, H., Grnbauer, M., Proost, R., Veringa, H., and Meyer, J. J. C. (2014). *Enhancing Products by Embedding Agents: Adding an Agent to a Robot for Monitoring, Maintenance and Disaster Prevention*, volume 449 of *Communications in Computer and Information Science*. Springer Berlin Heidelberg.

Paolucci, M. and Sacile, R. (2005). *Agent-based manufacturing and control systems : new agile manufacturing solutions for achieving peak performance*. CRC Press, Boca Raton, Fla.

Ruta, M., Scioscia, F., Loseto, G., and Di Sciascio, E. (2012). An agent framework for knowledge-based homes. *Third International Workshop on Agent Technologies for Energy Systems (ATES 2012). A workshop of the Eleventh International Conference on Autonomous Agents an Multiagent Systems (AAMAS 2012)*.

Song, J., Song, H., Mok, A., and Chen, D. (2008). Wirelesshart: Applying wireless technology in real-time industrial process control. *Symposium on Real-Time and Embedded Technology and Applications*, pages 377–386.