Making and Understanding
A Vision for IoT Makerspaces

Julian Dax and Volkmar Pipek
Institute for Information Systems, University of Siegen, Siegen, Germany

Keywords: Making, IoT, Embedded Devices, End-User Development, End-User Software Engineering, Infrastructuring.

Abstract: We present a vision for the IoT makerspace of the future. Currently, makers design their spaces with a focus on building (or making), but the core challenge they face in the IoT era is understanding. In our vision this is archived by gathering data about the IoT devices and their environment, storing that data in a central repository, consolidating it and making it easily accessible. We also describe the first steps we took towards this vision.

1 INTRODUCTION

Consider the following scenario: Alice is building a small robot in her local makerspace, which automatically drives towards the nearest light source. Two weeks ago, she almost got it working: It could follow a light source, but the wheels don't work well on smooth surfaces. Now, that Alice has time again, she wants to fix that. After she puts it on the big working table in the hackspace and shines towards it using her flashlight it does not move one bit. She is puzzled and asks herself what she did do differently this time.

Problems like these are quite common and illustrate several concrete challenges makers face every day. Software-controlled devices like the robot behave in complex ways. Understanding the behavior of a robot is therefore very challenging. As an IoT device, is does not only have internal complexity; it also reacts to the environment. In the robot's case, this environment comprises its light sensors. In IoT in general, it can be all kinds of sensors and also the network. With IoT, the challenge is not building anymore - the challenge is *understanding* the behavior of the things you are building.

In this position paper, we present a vision of the IoT makerspace (IoTMS) of the future, where the IoTMS *itself* is an infrastructure which provides tools for understanding. Concretely this means that:

- The IoTMS captures and stores sensor data automatically.
- The IoTMS allows Makers to run simulations of the developed IoT devices.
- The IoTMS supports data analysis, retrieval, and sharing.

We will also describe the first steps we took to arrive at such a future by implementing a prototype application.

2 REQUIREMENTS FOR AN IoT MAKERSPACE

Back to Alice's robot problem: After two hours of checking the cables, debugging the source code and replacing the motor she realizes, that when she last worked on her project it was a Tuesday evening, and now it is a Sunday afternoon. The lighting conditions changed, and her algorithm has to change with them. After some trial and error and a lot of recompilation, she finds a setting for the light sensors which works in the afternoon sun.

How can Alice be supported in understanding what her robot does? At a very basic level, it is a matter of gathering data. All sensor input and network traffic should be directly visible to her. In general, we derive requirement R1 from this:

- The current state of the IoT device and its environment should be visible to makers at all times.

Fulfilling this requirement will help Alice see previously hidden information which her robot

Dax, J. and Pipek, V.
Making and Understanding - A Vision for IoT Makerspaces.
DOI: 10.5220/0006094603140317
In *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016) - Volume 3: KMIS*, pages 314-317
ISBN: 978-989-758-203-5

depends on. In this case, making visible the previously invisible light sensor reading will help her realize, that she needs to change her algorithm to reflect the different input values. This will help with many problems in the makerspace. However, Alice's predicament is an exceptionally complicated one. She can only solve it by also taking into account data from her last visit to the space because if she adopts her algorithm now to working in the afternoon, she also wants to know which light sensor readings were taken in the evening or her algorithm will not operate under all circumstances. Abstractly, we derive requirement R2 from this:

- The past states of the IoT device and its environment should be visible to makers at all times.

Even if she could see the state of her robot and its environment in past and the present, Alice would still have a problem: She cannot test, if her updated algorithm would work in the dark or if she just broke it for that use-case. What would the robot do, if there would be different lighting conditions? This question can be addressed by allowing the simulation of different scenarios.
We derive Requirement R3 from this:

- Makers should be able to simulate environment and device state parameters.

After she had finished her robot, Alice moved to another city. Half a year later, Bob is interested in building his own light-following robot. During the project, he runs into the same problems Alice faced before. However, he is not aware that someone in the same makerspace built the same project as he does. From this, we derive two requirements, R4 and R5. Firstly, Alice should be supported in capturing knowledge and secondly, Bob should be supported in retrieving it:

- IoTMS should help makers proactively when they run into problems by providing information relevant to the current context.
- Makerspaces should support makers in reviewing the gathered data and procure material for future reference from it.

3 THE VISION

We envision an integrated hard- and software system, which is distributed throughout and also an integral part of the makerspace. We also see the IoTMS as one holistic system. It is an infrastructure which provides tools for building and understanding IoT devices.

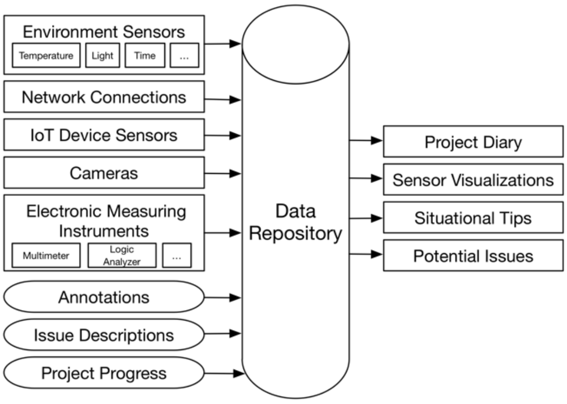

Figure 1: Overview of the data-flow through the IoTMS.

To address R1, it builds on connected sensors. For electronics, this can be standard tools like oscilloscopes and multimeters which transmit their readings to a central data repository. Equally important is reusing the already existing sensors in IoT devices. In the robot example, the robot should not only use the light sensor internally; it should send all gathered data to the repository as well. Moreover, we envision that computer vision technology can automatically identify electronic components, like resistors or diodes. After the sensors capture the data, the system visualizes it. For that makers can use their own laptops but the space also comes with projectors or big screens. Makers should be able to visualize the data by picking from a set of pre-defined visualizations. The data repository and data visualization also address requirement R2. A modern time series database which allows quick access to all captured information would allow makers to search for past sensor data. To allow the simulations described in R3, the software running on the IoT device under development would also have to run in the simulator. For that, the system should emulate or simulate the microcontroller used in the IoT device.

Lastly, there would be a system for annotating the captured data with the current project (for example "Building a light-following robot"), project progress and issues the maker faced. This way, a lab diary is automatically generated. This diary can help makers who do the same or similar projects in the future. For this, we envision a context-dependent ambient learning system which assists makers with their concrete problem. The system could analyze current sensor- and metadata data and automatically find similar situations in the past using clustering algorithms. Based on these past situations, the system could then provide the maker with tips and point out possible issues.

4 FIRST STEPS TAKEN

As a first step towards the vision just described, we implemented a desktop application called ‘Remotino’ (Dax et al., 2016) using web technologies (Electron, Redux, React) which allows makers to remotely control and instrument Arduino microcontrollers.

As shown in figure 2, Remotino allows makers to visualize analog and digital inputs to the microcontroller. Data from the Arduino is transferred via USB to the desktop application using the Firmata protocol (Steiner, 2009) . The main aim of the app is to make the current state of the Arduino visible to the user (R1).

Figure 2: A screenshot of the Remotino tool, showing several analog inp ut pins and visualizing one of them.

It also provides some information about the past state of the Arduino (R2), as the visualization uses a sliding time window of one minute. Besides the input visualization, Remotino also shows all available pins and which modes these pins support (digital in, digital out, analog in, analog out). It also recognizes which kind of Arduino is connected. All these features aim to make information visible and understandable which was previously invisible.

Remotino is a first step towards the IoTMS and currently only implements some of the aspects of a IoTMS for a single maker. It only runs locally on a PC and does not connect to a central repository (like described in figure 1). It also only works with the Arduinos it is directly connected to via USB. To address these issues and develop it into the basis of an IoTMS we are currently working on the first version of the central data repository described above and on an integration between this repository and Remotino.

Figure 3: A screenshot of the Remotino tool, showing the two Arduinos which are automatically detected when plugging them into the PC.

5 RELATED WORK

In our work, we draw on three research areas: Making, Infrastructuring and End-User-Development (EUD) (Ko et al., n.d.; Lieberman et al., 2006).

In the making and personal fabrication area, Sheridan et al. describe and analyze current practices in three makerspaces (Sheridan et al., 2014). Mellis and Buechley studied DIY-communities with a focus on online communities and electronic products. They emphasize the need for “new Forms of knowledge transfer" and find that in online communities about DIY electronics text-based communication in a question-and-answer format is very common but ineffective (Mellis and Buechley, 2012).

In EUD, there has recently been more interest in the development of physical objects and making. Booth et al. identified challenges, which makers face when building IoT devices (Booth et al., n.d.).

In infrastructure research (Star and Ruhleder, 1996) there has been interest in how to design infrastructures, which help people in the appropriation of technology (Ludwig et al., 2014). We see the IoTMS as an infrastructure for collaboratively understanding and building IoT.

REFERENCES

Booth, T., Stumpf, S., Bird, J., Jones, S., n.d. Crossed Wires: Investigating the Problems of End-User Developers in a Physical Computing Task. ACM, New York, New York, USA.

Dax, J., Ludwig, T., Pipek, V., 2016. Remotino: Supporting End-User Developers in Prototyping Embedded Devices CEUR Workshop Proceedings, Bari, Italy, pp. 7–11.

Ko, A. J., Abraham, R., Beckwith, L., Blackwell, A., Burnett, M., Erwig, M., Scaffidi, C., Lawrance, J., Lieberman, H., Myers, B., Rosson, M., Rothermel, G., Shaw, M., Wiedenbeck, S., n.d. ACM Computing Surveys (CSUR) 43, 21–44.

Lieberman, H., Paternò, F., Klann, M., Wulf, V., 2006. End-User Development: An Emerging Paradigm, in: Lieberman, H., Paternò, F., Klann, M., Wulf, V. (Eds.), Springer Netherlands, Dordrecht, pp. 1–8.

Ludwig, T., Stickel, O., Boden, A., Pipek, V., 2014. Towards sociable technologies: an empirical study on designing appropriation infrastructures for 3D printing. ACM.

Mellis, D., Buechley, L., 2012. . Journal of the Designing Interactive Systems Conference.

Sheridan, K., Halverson, E., Litts, B., Brahms, L., Jacobs-Priebe, L., Owens, T., 2014. . Harvard Educ Rev 84, 505–531.

Star, S., Ruhleder, K., 1996. . Information Systems Research 7, 111–134.

Steiner, H.-C., 2009. Firmata: Towards Making Microcontrollers Act Like Extensions of the Computer. pp. 125–103.