HomeLab: Shared Infrastructure for Home Technology Field Studies

A.J. Bernheim Brush, Jaeyeon Jung, Ratul Mahajan, James Scott
Microsoft Research
{ajbrush, jjung, ratul, jws}@microsoft.com

ABSTRACT

Researchers who develop new home technologies using connected devices (e.g. sensors) often want to conduct large-scale field studies in homes to evaluate their technology, but conducting such studies today is quite challenging, if not impossible. Considerable custom engineering is required to ensure hardware and software prototypes work robustly, and recruiting and managing more than a handful of households can be difficult and cost-prohibitive. To lower the barrier to developing and evaluating new technologies for the home environment, we call for the development of a shared infrastructure, called HomeLab. HomeLab consists of a large number of geographically distributed households, each running a common, flexible framework (e.g., HomeOS [4]) in which experiments are implemented. The use of a common framework enables engineering effort, along with experience and expertise, to be shared among many research groups. Recruitment of households to HomeLab can be organic: as a research group recruits (a few) households to participate in its field study, these households can be invited to join HomeLab and participate in future studies conducted by other groups. As the pool of households participating in HomeLab grows, we hope that researchers will find it easier to recruit a large number of households to participate in field studies.

Author Keywords Home automation, smart home, domestic technology, devices.

ACM Classification Keywords H.1.2 User/Machine systems, H.5.2 User Interfaces.

General Terms Human Factors

INTRODUCTION

Several studies, many focusing on home networks, have described the challenges that households face setting up, managing and using existing technology in their homes [e.g. 3, 6, 7, 13, 15, 19, 21, 23, 24]. These challenges include the effort necessary to manage systems, diversity across households' technological setups, and varying household routines. More generally, as researchers develop new home technologies to support a range of goals from aging in place to social connectedness to sustainability, they face challenges evaluating the effectiveness of their prototypes, which typically involve deploying some type of connected device (e.g. sensors) in realistic settings.

User studies of such prototypes conducted in laboratory settings, while valuable in initial stages to collect feedback, lack the realism necessary to truly understand if a system will work in an actual home setting. Some institutions have built smart home laboratories where participants can reside and participate in longer term studies (e.g., Aware Home [12], House_n [11], Orange [20], Tampere [14]). Smart dorms support gathering longer term data from students, for example, Duke University's Smart Dorm, a co-ed residence for 10 students [5]. While these home and dorm laboratories allow for longer term deployment, the studies are still conducted with a relatively few number of participants under controlled conditions.

When researchers do seek to deploy their prototypes into homes, logistical constraints typically limit them to deploying into a small number of homes in the same geography. This may be completely appropriate when the research and the prototype are in an early phase where the main goal is often to quickly get something in use and gather feedback to guide refinements. But after this phase, the intent of the researchers is often to deploy, evaluate, and study their prototype at a larger scale to gather feedback from more households to validate, disprove or extend initial (possibly biased) findings.

It is these large-scale deployments that are extremely challenging today. They require considerable engineering effort to ensure hardware and software prototypes work robustly with each household's unique infrastructure. Unfortunately, lacking a common framework for prototype development, individual research groups tend to do this engineering almost from scratch and their work is not easily reusable by other groups. Further, recruiting households and deploying hardware take considerable effort and money, making it difficult to expand beyond the immediate geographic reach of the research group. In theory, these challenges can be overcome, but in practice they significantly raise the barrier for experimental work in this domain and limits field studies to a small number of households.

How can we reduce the effort and expertise needed to conduct large-scale and geographically diverse field studies of home technology? Towards this goal, we propose the development of a shared infrastructure, which we call HomeLab. Our proposal is inspired by the success of PlanetLab [18] which enabled development and evaluation of global network services.

Our vision is that HomeLab consists of a large number of homes in various parts of the world. Each home runs a common framework in which experiments are written. This framework helps decouple experiment-specific logic from the more general mechanisms to robustly interact with connected devices in the home. This decoupling enables research groups to share engineering effort, experience, and expertise. This lets them focus more of their time on tasks specific to their research goals (e.g. health, sustainability) or novel devices rather than spending time and effort developing their own infrastructure from scratch. As researchers extend HomeLab with additional capabilities these additions would be available to others. The task of growing the infrastructure to include more homes also gets distributed and shared across research groups, as individual groups help recruit a subset of the households. We hope a large pool of households in

UbiComp'12, September 5-8, 2012, Pittsburgh, USA.
ACM 978-1-4503-1224-0/12/09.

HomeLab would allow researchers to more easily expand their experiments to a many, diverse homes.

While developing an infrastructure like HomeLab might seem challenging, we believe that it is feasible. In recent work, we have developed a platform called HomeOS, which presents to users and developers a PC-like abstraction for technology in the home and simplifies the tasks of writing applications and managing devices [4, 8]. This platform has been used by over twenty research groups to develop applications for the home on top of a common set of mechanisms to interact with devices. These groups have also been able to develop additional mechanisms to communicate with new types of devices in ways that those mechanisms can be reused by other applications. HomeOS offers a potential starting point to developing HomeLab.

But we must address several additional challenges if we are to develop a robust, sustainable experimental infrastructure that is shared by the community. These challenges are technical (e.g., how to isolate experiments, and how to provide enough information for researchers to diagnose unexpected behaviors in a remote deployment) as well as legal (e.g., preservation of intellectual property and safety of participating households) and social (e.g., incentives for households to participate and research groups to recruit homes). We discuss these challenges in the body of this paper.

Our goals in writing this paper are two-fold. The first is a call to arms for the research community; it is time for us to collectively figure out way to facilitate large-scale field studies in homes. The HomeLab vision we describe offers one potential approach, and we hope that this paper helps fuel a broader discussion on other promising approaches. Assuming HomeLab is a promising approach, our second goal is to get feedback from the community on how to make it a reality and invite interested members to get involved.

PROBLEM AND MOTIVATION

Today, the barrier is high to developing and evaluating new technologies for the home environment. Our particular focus is on technologies that involve interacting with one or more connected devices in the home, such as motion sensors, thermostats, light switches, energy meters, cameras, speakers, etc. These devices are necessary building blocks for a range of scenarios across energy management, security, awareness, comfort, convenience, and health. However, not all kinds of home technologies rely on connected devices. One example is providing visibility and control over home network traffic [1, 2, 9, 10, 17], for which several research groups have conducted large-scale studies. While networking monitoring studies could be conducted using HomeLab, they are not our primary focus. Instead, our goal is to enable extended deployments (e.g. weeks, months) where participants live with prototypes that include connected devices.

Research that involves connected devices suffers from two key problems today, which we outline below. As a concrete illustration of these problems, we draw upon recent experience of deploying a technology called PreHeat whose goal was to efficiently heat homes by using occupancy sensing [22]. The experience is representative of challenges faced in our other field studies, and we believe these challenges are also faced by other researchers deploying technology in homes.

Large engineering effort, not easily reused: Developing a prototype for a home technology involves a huge amount of engineering effort towards robust means to interact with devices of interest. This effort often goes well beyond what is needed to test the primary research hypothesis. For PreHeat, the primary research goal was to develop algorithms to learn a home's occupancy pattern and experimentally quantify these algorithms' effects on energy consumption and comfort, in both whole-house and per-room heating scenarios. This necessitated installing new sensors and actuators, but also providing a complex management "layer" to the system for communicating between the distributed hardware elements, hosting the controlling software on a PC, processing and storing the data generated by the sensors, providing a high "uptime" through failure recovery mechanisms, providing a means for researchers to configure, remotely monitor and remotely update the system, and so on. That this effort is needed is not a problem by itself—prototype development in other domains too often involves engineering that is not central to the primary research question—but unfortunately the engineering effort is not easily reused by other research groups.

The difficulty of reuse stems from the fact that what is implemented is coupled directly with the realization of the immediate research goals. This coupling is understandable as the researchers are most focused on their immediate goals and make pragmatic decisions during development. It is also unavoidable because the community lacks a common framework to implement their research prototypes in a way that enables reuse.

Some research projects also need to engineer custom hardware. For PreHeat, we evaluated a variety of off-the-shelf devices such as motion sensors, but ended up building our own devices due to a variety of issues including lack of access to fine-grained sensed data (rather than summaries). This took a number of months to prototype and manufacture, which requires hardware skills that limit who conducts such research in practice. Fortunately, this barrier to entry is being eroded through the availability of device prototyping platforms. We discuss later how we can use such platforms to reduce and share engineering effort for research based not only on commodity devices but also custom devices.

Limited deployments, often without geographical diversity: The second problem is that scaling beyond a small-sized deployment is incredibly challenging. The challenge stems from two factors. The first one is logistical—it is hard to recruit and support participants that are not within immediate geographic reach of the research team. Recruiting volunteers is not simply a matter of shipping them the necessary hardware and software, but it often needs hardware setup and training that can require personal interaction. This factor was one of the reasons why a large-scale deployment of PreHeat was not conducted. Most research deployments are similarly small (less than a dozen) and not geographically diverse.

The second factor is cost. Because hardware is involved, the total cost of deployment scales linearly with the size of the deployment. For instance, PreHeat deployments in addition to the hardware cost of the thermostat and actuation devices required a dedicated PC, which interacted with devices and ran the learning and control algorithms, since no infrastructure was in place to allow use of a PC shared with other applications. The combined cost of these devices is another hurdle in the way of a large-scale deployment of PreHeat.

HOMELAB VISION

To address the problems of non-reusable engineering effort and small-scale deployments, we call for the development of a shared research infrastructure. Multiple research groups contribute to this infrastructure and each is able to leverage the investments of others. We imagine that this infrastructure, which we call HomeLab (see Figure 1), is deployed across many homes, though each home has different hardware and devices, including custom devices.

Households join HomeLab either as part of a deployment “Managed” by a research group or as an “Unmanaged” household. Managed homes are recruited by a research group participating in HomeLab and that group is responsible for keeping parts of the infrastructure in those homes in a functional state and up-to-date. These homes are likely, but not necessarily, in their geographic locality.

Unmanaged homes have a motivated “Do-it-yourselfer (DIY)” willing to provision, install, and manage the infrastructure and could apply to participate in HomeLab. We plan to include Unmanaged homes in HomeLab because, on presenting PreHeat or HomeOS research to many audiences, a common question is “I have done some home automation work – can I try this out for myself?” Such DIY users could provide valuable feedback on a research system, but in most cases we have to turn them down because the study in question has already started (or even ended), or they are ineligible for another reason (out of locale, or “too expert”), or simply that with managed studies one cannot accept unlimited participants. But such enthusiastic users could be valuable for many experiments and the overhead for their inclusion into HomeLab is lower than that for Managed homes.

At any given time, multiple experiments, belonging to different research groups, are active on HomeLab. Experiments are expressed in a common, flexible framework to enable reuse and remote control. Each home volunteers to participate in one or more active experiments, based on the residents’ interest and the devices that the home contains. If the residents interested in an experiment do not have the needed hardware, the experiment owners can ship them the device. Once the original experiment completes, other research groups can reuse it easily for their experiments, perhaps after contributing to the original cost of the hardware. Additionally, other research groups can also reuse the software used to interact with devices. Alternatively, experiment owners might mandate that a device be returned after the experiment or allow a household to purchase the device. Allowing HomeLab households to keep the device makes it available for future experiments, while allowing the original experimenter to recoup some of their investment.

Assuming this vision can be realized—we discuss in the next section how that might happen and the challenges involved—we can see how HomeLab addresses the problems mentioned in the previous section. Research groups are able to reuse each other’s engineering towards interacting with devices and amortize the cost of hardware over many experiments. Instead of needing to recruit all homes themselves, they will have access to a diverse set of homes that have been recruited by other research groups (Managed) or independently enthusiastic (Unmanaged). For PreHeat, if HomeLab were available, software for interacting with thermostats and occupancy sensors might have already existed or if not, once written the software could have been more easily reusable by others. Expanding the deployment to a large number of homes might have been able to leverage existing hardware rather than requiring a PC, thermostat, and occupancy sensors for each additional home.

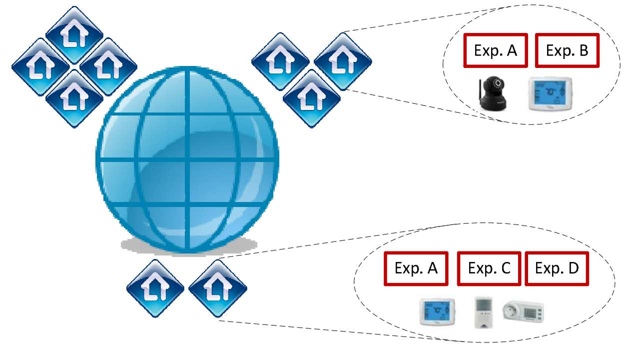

Figure 1: HomeLab would consist of multiple sites across the world. Each site has multiple homes and is deployed by a research group. Each home runs one or more experiments and contains one or more devices needed for the experiment.

Further, HomeLab also lowers the barrier for technical expertise needed for research in home technology. Today, the research groups need expertise in not only their core area of interest (e.g., health care or energy management) but also in developing low-level device interaction software. If software for interacting with devices of interest already exists in HomeLab, research groups can focus on their core interest.

REALIZING HOMELAB

In this section, we outline how the HomeLab vision can be made a reality and discuss many open questions. We first describe how HomeOS could enable HomeLab and then the additional functionality needed to support running experiments in HomeLab, as well as open non-technical issues that will need to be resolved to create HomeLab.

HomeOS as a Starting Point

A primary requirement for HomeLab is a common platform for implementing technology. This platform should provide a standard way to communicate with devices in all homes where those devices exist. So, if controllable thermostats exist in multiple homes, they should be controllable in a similar manner by experimental software that wants to control them, independent of the thermostat vendor and communication protocol (e.g., Z-Wave or ZigBee). This requires abstracting the details of individual devices and offering higher-level APIs based on device type for experimental software development.

Over the past two years, we have developed a platform called HomeOS (see Figure 2) to enable this separation between devices and higher-level software [4]. In HomeOS terminology, the software modules that communicate with devices are called drivers and higher-level control software modules are called applications. HomeOS drivers communicate to devices using protocols that are specific to the device but expose to applications high-level APIs that depend on the type of the device and its functionality. For instance, the API for a light includes “on” and “off” commands. HomeOS applications are written using the high-level APIs exposed by the device drivers. They do not embed in them any knowledge of the device protocol or vendor.

HomeOS currently runs in 12 homes in the Pacific Northwest region of the USA. It has support for many types of devices such as light switches, dimmers, door/window sensors, and cameras. We have also written eighteen applications that use these devices in various ways.

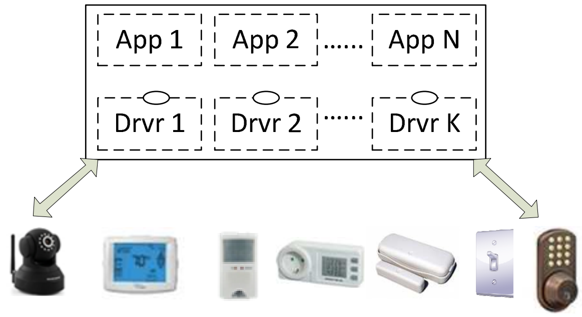

Figure 2: HomeOS is a platform that runs on a PC. Software modules called drivers communicate with devices and offer to applications high-level APIs to sense and actuate those devices.

We have made the HomeOS prototype available to academic institutions to encourage teaching and research on connected homes and devices. Over 50 students across twenty research groups have developed applications and drivers for HomeOS (see [8] for examples). Applications include energy profiling, remote monitoring, and end user programming; and drivers include those for cameras, energy meters, and certain kinds of ZigBee devices. Some of these research groups are also engaged in deploying HomeOS for their research needs. Sharing device drivers sometimes happen through personal communication, but they are not able to scale their deployment.

We envision HomeLab to be based on HomeOS. The participating homes have a dedicated PC that runs HomeOS, along with drivers and applications that are relevant to the experiments in which they chose to participate. They may also run other HomeOS applications for their own use (e.g., remote access to the home cameras) that are not part of any experiment.

To run an experiment in HomeLab, research groups would create (or convert if it already exists) their research prototype as a HomeOS application. Given a rich set of devices that HomeOS currently supports, we do not expect that this effort to be any more difficult than developing a Windows application. However, some experiments may require new sensing devices whose device driver is not yet implemented as part of HomeOS. In such cases, research groups may have to develop required device drivers themselves and need special permission to upgrade participants' HomeOS to include the new drivers. Although this entails additional effort, we hope that research groups that participate in HomeLab will share driver code, which means that other groups will not have to write a driver for these devices.

Once an experiment is ready to deploy as a HomeOS application, next steps include recruiting participants, running the HomeOS application remotely in participating homes, deploying any custom devices necessary, and collecting data. In the next section, we discuss technical challenges that may arise in each step. Furthermore, While HomeOS provides a good starting point; several other pieces are needed to fully realize the HomeLab vision. The remainder of this section discusses these pieces.

Supporting Custom Devices

Some research projects need custom devices. While this need is not universal, completely avoiding custom devices limits the types of research HomeLab can enable. Fortunately, recent innovations in prototyping platforms are making it easier to build custom devices for HomeLab. The underlying platform used in the PreHeat hardware, Microsoft .NET Gadgeteer [16] is now an open hardware standard, and multiple manufacturers are retailing over eighty hardware modules. Other platforms in this space include Twine [25], which makes it easy to interface sensors to a cloud-based service and provides simple rule-based programming to define actions based on particular sensed events, and the well-known Arduino platform.

The use of such platforms simplifies the challenge of creating custom devices (e.g., no experience with circuit schematics or soldering irons is needed). Further, HomeOS makes it easy to integrate devices that are based on such platforms. A base software module can be written once that is specific to the platform (Device Connectivity Layer in HomeOS terminology). Additional devices then only need a simpler driver that is specific to the device functionality.

To enable HomeLab, we propose to extend HomeOS to support a few popular platforms. Research groups that need a custom device, create the device using one such platform (or extend HomeOS to a new platform) and write a small device driver to interact with the device. Alternatively, research groups can also create devices from scratch without using any hardware platform, but the engineering effort will be bigger in such cases.

Additional Software Capabilities

From a technical point of view, HomeLab is a service running on top of a network of HomeOS machines. However, unlike other shared experimental platforms such as PlanetLab [18] in which each participating machine is similarly configured, participating homes will differ greatly from types of devices installed, network configuration, to system reliability. There may also be limited administrative support for installing, executing, and debugging experiments depending on whether a home is a `Managed` or `Unmanaged` household and on the expertise of the technology guru in the house. In the light of these challenges, below are three technical properties that we view important for HomeLab.

Automatic configuration discovery of participating homes: An experimenter may want to "screen" available HomeOS machines for particular conditions such as what devices are installed and how these devices are used by other applications. To help this recruiting process, HomeLab should automatically discover such conditions of participating HomeOS machines and make them available to experimenters in a privacy-preserving manner. Although a list of screening conditions may grow over time, useful system properties to monitor include runtime device configuration, system uptime, and network configuration and usage. HomeLab must also provide a mechanism for experimenters to advertise their experiments to candidate homes so that participating households can remain anonymous unless they opt to participate in an experiment.

Isolation of experiments: Experiments should not interfere with existing HomeOS applications running in the participant's home or with other experiments. The current HomeOS system supports the isolation of application-level modules and restricts their interactions with the core system through the system APIs. However, since experiments may compete with the existing applications for resource usage (e.g., network bandwidth, access to speakers), we need a capability to monitor and to restrict resource usage by HomeOS applications. This capability can be implemented by extending the kernel of HomeOS to support performance isolation, in addition to its access control facilities.

In addition to performance isolation, we need a capability to enforce security isolation against possibly malicious HomeOS applications. Currently, in a rare case that the HomeOS kernel is compromised, malicious applications can "own" all the devices and experiments running in the compromised system. We also need a capability to protect the privacy of experimenters' code and data. Although the application-level isolation provided by HomeOS prevents "curious" applications from accessing data or code belonging to other applications running in the same system, we may need to support cryptography mechanisms such as encrypted storage for stronger protection.

While technical isolation is critical, ensuring good experimental design is also necessary. It may be possible to run some experiments in parallel in a house, but many others may conflict. HomeLab must include mechanisms to make sure that research groups currently running an experiment in a home are consulted before a house can participate in additional experiments. Over time, through experience, we hope that we can develop means to programmatically specify what does or does not interfere with their experiments. This specification then becomes part of the experiment manifest and is consulted when a research group is looking for homes that are eligible for a new experiment.

Remote management and diagnostics: Experimental software will likely need to be remotely configured and updated in a well-controlled fashion; for example, there may be multiple phases to the experiment for comparative evaluation purposes. Experimental software may also crash or malfunction when running in the participant's HomeOS. Since experiments have limited visibility into the runtime environment and device status, providing logs of how experiments interacted with the system can be useful for debugging. In particular, in the component-based system like HomeOS, coredump alone may not suffice as errors could be caused by unexpected interactions with other system modules. One approach to help this type of system-level debugging is to make the interaction logs with devices be detailed enough that the experimenters can reproduce the conditions in their lab once failures occur in the wild.

Further, to track the progress of experiments, it would be useful for researchers to have an online view of status that crosses houses rather than browsing each house individually—showing configuration status, key experimental metrics in real time, failure/reliability metrics, and so on.

Other Open Questions

Realizing HomeLab also entails non-technical issues such as providing the legal protection of experimenters' software and creating the right incentives for users to participate in HomeLab and for research groups to recruit homes and contribute hardware and software. We briefly discuss some of the issues in this section.

Incentives for experimenters to share technology, code and study sites: We hope that research groups will voluntarily share reusable code with the community to collectively improve HomeLab. As there are many standard ways to maintain a large code base with distributed developers, it will require little technical efforts to facilitate code sharing. However, it may require stronger social incentives to encourage researchers to share code and possibly technology. Ensuring HomeLab runs smoothly may require some "fair trade" policies determined by the community using HomeLab and adapted as necessary. For example, PlanetLab established a system where participating research institutions had to contribute a site (or pay money). Thus, as more institutions wanted to use PlanetLab for their research, the infrastructure naturally scaled with the level of interest.

In HomeLab, perhaps research groups can earn credits for different types of contributions, such as creating drivers for new devices, deploying hardware that can be re-used, or recruiting and managing a number of households. This type of system could support research groups having different strengths, as people that build novel devices might still want to deploy them, but not have experience recruiting households. We expect these mechanisms will need discussion and iteration to be successful. Our core goal is to discourage free-riding and achieve rough fairness; people should benefit in proportion to the effort that they contribute.

Incentives for continued participation in HomeLab: There are two ways a household could join HomeLab: as a Managed deployment or as Unmanaged home. In the first case, a research group might recruit a house to be part of a Managed deployment for a specific experiment. For this house, the installation of HomeOS and any required devices would be managed and supported by the research group. We expect for the duration of that experiment the household would participate only the experiment of the research group that recruited the household and the opportunity to recruit the household into HomeLab would come at the end of the first experiment. At this point, the research group could then describe HomeLab, give examples of other experiments and hopefully convince the household to join HomeLab.

We are aware that encouraging homes to join HomeLab may be challenging since they are being asked to consider participating in experiments conducted by researchers they do not know. However, in our own field studies we have been able to recruit families unknown to ourselves that were interested in trying the novel technology we had built and then build a trust relationship with those households. So we hope some of the Managed households will be interested in participating in HomeLab and experiments conducted by other researchers.

The possibility of financial incentives frequently offered in user studies may also help entice homes to join HomeLab. Our own group is interested in creating logging "experiments" open to all HomeLab households with raffle incentives and other compensations that may appeal to households. Retaining Managed households in particular is important because we expect fewer of them to be technology enthusiast households compared to households that may apply directly to join HomeLab as Unmanaged homes. Of course, another way to retain houses in HomeLab is to ensure the platform provides services that a household wants, e.g., energy monitoring or cheap home security.

We may also explore requiring households running HomeOS to participate in some number of experiments as "payment" for using the platform. For households that apply to be Unmanaged homes, out of a desire to use HomeOS in their own homes, this type of approach could work well. We hope that the 12 homes already running HomeOS will opt into participating in HomeLab because they are already experienced with HomeOS.

Household data privacy and informed consent: Experiments conducted in people's homes are likely to collect data that raise privacy concerns (e.g. occupancy history, media or energy usage). When researchers are recruiting households from HomeLab to participate in experiments there needs to be some mechanism to ensure that these experiments are being conducted ethically so that participants know exactly what data is being collected and can opt-out at any time. We expect that each research group that participates in HomeLab already has a human subjects reviews process required by their organization and those requirements would help ensure appropriate care is taken with participants' data. However, because requirements may vary by institution, we expect the community participating in HomeLab will need to come to an agreement around additional mechanisms to ensure that experiments are appropriately conducted. For example, potentially establishing a standard format for a consent form telling households what data that would be collected and how it would be used. An important direction for future work is investigating how the platform itself can help validate that data access and handling actions match what is specified in the application's manifest.

Legal issues: Research groups may wish to protect the intellectual property of the software to be deployed in HomeLab. Since requiring a legal contract with study participants, especially those who live in remote location (that may part of different legal jurisdiction) is costly, HomeLab can provide technical solutions such as a code signing if a need for digital rights management for experimenters' software arises. HomeLab needs to maintain signing keys for experimenters and HomeOS needs to be extended to support basic cryptographic operations.

CONCLUDING REMARKS

Creating HomeLab is an ambitious goal that will require the collective effort of many research groups and households to be successful. While many challenges remain unanswered, we are excited and optimistic about such a shared infrastructure for home technology field studies. We also look forward to discussing and debating the merits of HomeLab and the best way to realize it with other researchers in this area.

ACKNOWLEDGMENTS

We thank the people who have used and extended the HomeOS system. We also appreciate feedback from the workshop reviewers which helped us improve the paper.

REFERENCES

1. BISmark, http://projectbismark.net/
2. Chetty, M., Haslem, D., Baird, A., Ofoha, U., Sumner, B., Grinter, R., Why is My Internet Slow?: Making Network Speeds Visible. Proc. CHI 2011,1889-1898.
3. Chetty, M., Sung, J-Y., Grinter, R., How Smart Homes Learn: The Evolution of the Networked Home and Household. *Proc. UbiComp 2007*, 127-144.
4. Dixon, C., Mahajan, R., Agarwal, S., Brush, A., Lee, B., Saroiu, S., Bahl, V., An Operating System for the Home, *Proc. NSDI 2012*.
5. Duke's Smart Dorm, http://smarthome.duke.edu/dorm Adobe Acrobat Reader.
6. Grinter, R., Edwards, W., Chetty, M., Poole, E.S., Sung, J.-Y., Yang, J., Crabtree, A., Tolmie, P., Rodden, T., Greenhalgh,C., Benford, S., The ins and outs of home networking: The case for useful and usable domestic networking. *ACM ToCHI 16*, 2(2009), 8:1-8:28.
7. Grinter, R., Edwards, W., Newman, M., Ducheneaut, N., The Work to Make a Home Network Work. *Proc. ECSCW 2005*, 469-488.
8. HomeOS, http://research.microsoft.com/en-us/projects/homeos/default.aspx
9. HomeNet Profiler, http://cmon.lip6.fr/hnp/pages/home
10. Homework Research project, http://www.homenetworks.ac.uk/
11. Intille, S. Designing a home of the future. *IEEE Pervasive Computing 1*, 2(2002), 80-86.
12. Kietz, J., Patel, S., Jones, B., Price, E., Mynatt, E., Abowd, A., The Georgia Tech Aware Home. *Ext. Abstracts CHI 2008*, 3675-3680.
13. Kim, T., Bauer, L., Newsome, J., Perrig, A., Walker, J., Challenges in Access Right Assignment for Secure Home Networks. *Proc. HotSec 2010*.
14. Koskela, T., Väänänen-Vaninio-Mattila, K. Evolution towards smart home environments: empirical evaluation of three user interfaces. *PUC 8*, 3-4(2004), 234-240.
15. Mazurek, M., Arsenault, J., Breese, J., Gupta, N., Ion, I., Johns, C., Lee, D., Liang, Y., Olsen, J., Salmon, B., Shay, R., Vaniea, K., Bauer L., Cranor, L., Ganger, G., Reiter, M., Access control for home data sharing: Attitudes, needs and practices. *Proc. CHI 2010*, 645-654.
16. Microsoft .NET Gadgeteer, http://netmf.com/gadgeteer/
17. Netalyzr, http://netalyzr.icsi.berkeley.edu/
18. Peterson, L, Anderson, A., Culler, D., Roscoe, T., A Blueprint for Introducing Disruptive Change in the Internet. *Proc. HotNets 2002*.
19. Poole, E., Chetty, M., Grinter, R., Edwards, W.K., More than Meets the Eye: Transforming the User Experience of Home Network Management. *Proc. DIS 2008*. 455-464.
20. Randall, D. Living Inside a Smart Home: A Case Study. In: Harper, R. (ed.) Inside the Smart Home, 227-246. Springer, Heidelberg (2003).
21. Rode, J., Toye, E., Blackwell, A. The Domestic Economy: a Broader Unit of Analysis for End User Programming. Proc. CHI 2005, 1757-1760.
22. Scott, J., Brush, A., Krumm, J., Meyers, B., Hazas, M., Hodges, S., Villar. N., PreHeat: Controlling Home Heating using Occupancy Prediction. Proc. UbiComp 2011. 281-290.
23. Shehan, E., Edwards. W.K. Home Networking and HCI: What Hath God Wrought? Proc. CHI 2007, 547-556.
24. Tolmie, P., Crabtree, A., Rodden, T., Greenhalgh, C., Benford, S. Making the Home Network at Home: Digital Housekeeping. Proc. ECSCW 2007, 331-350.
25. Twine, http://www.supermechanical.com/