An Upcycled IoT:

Creating Tomorrow's Internet of Things Out of Today's Household Possessions

Kristin Williams

CMU-HCII-22-100
June 2022

Human Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15232 USA

Dissertation Committee
Scott Hudson (Co-chair), Carnegie Mellon University
Jessica Hammer (Co-chair), Carnegie Mellon University
Patrick Carrington, Carnegie Mellon University
Leah Buechley, University of New Mexico

A dissertation submitted in partial fulfillment of the requirements for the Doctor of Philosophy in Human Computer Interaction

This work was funded in part by National Science Foundation grant IIS-1718651.

Keywords: smart home, social objects, family collaboration, do-it-yourself, DIY, sustainability, IoT, internet-of-things, end-user programming, end user programming, personal inventories, attachment, intersectionality, paper computing, tangible interfaces, physical computing, interactive book, craft, stickers, RFID

Abstract

The Internet-of-Things (IoT) promises to enhance even the most mundane of objects with computational properties. Yet, IoT has largely focused on new devices while ignoring the home's many existing possessions. Requiring households to replace their possessions to adopt IoT creates substantial waste. This includes increasing artifacts diverted to the waste stream, as well as eroding agency, individual identities, values, and ways of life. To enable an alternative approach, this dissertation shows how IoT could augment existing household possessions rather than replace them. To do so, it worked with 10 American families to design an upcycled approach to IoT that makes use of existing household possessions and then built a system responsive to these findings. The results 1) describe patterns of families' socio-material practices, 2) developed techniques to enable existing possessions to be transformed with IoT services, and 3) presents The IoT Codex—a book of programmable and inexpensive, battery-free interactive devices—to support customizing everyday objects with software and web services using stickers. The presented work offers a lightweight approach to end user programming of everyday objects for customizing IoT to suit idiosyncratic socio-material practices.

Dedication

To Bunny Maureen Williams (1957-)—Mom: for teaching me I could get a degree at any stage in life. You inspired this work when you taught me to pick up my toys on the way to my bedroom. You reminded me that I should pick up after myself lest I get in the way of someone else's path to their own dreams.

Acknowledgements

This dissertation has been an adventure. I would not have made this extraordinary journey safely without the advice, support, and encouragement of an amazing group of people. For such a group, I am incredibly fortunate and grateful. While I cannot name all of those who have helped me along the way, I would like to specifically mention a few.

My advisers, Drs. Scott Hudson and Jessica Hammer, have been incredible guides along this adventure. Scott possesses the uncanny ability to both embrace the unvarnished realities of academic research and nurture some of the most fanciful and inventive ways of going about it. Similarly, Jess welcomes thorny and difficult social problems and yet, turns them into a game or some other playful form. I cannot thank the two enough for enabling my academic path and research on stickers. Thank you Scott and Jess for not only teaching me how to do research, but for making it fun along the way.

I had the fortune of having my thesis reviewed by two incredible minds. Drs. Leah Buechley and Patrick Carrington pushed me to develop the themes of this work more deeply and to see the greater possibilities this work could enable. Thank you Leah and Patrick for giving me the opportunity to learn from you and encouraging me to tackle challenging questions.

Carnegie Mellon comprises a community of faculty who are willing to let a wayward PhD student drop in their office with all kinds of questions. Thank you to the many faculty (some who have now moved on to other places) who helped me grapple with substantial challenges as I found my research questions and advisers: Carol Freize, Jeff Bigham, Amy Ogan, Geoff Kaufman, David Danks, Danielle Wenner, Aaron Steinfeld, and Anind Dey.

Queenie Kravitz is more than the associate director of the PhD program. I cannot thank her enough for the myriad of ways that she goes beyond her job and supports the whole person. Countless faculty and students would not be where they are today if it were not for her. I am among them. Throughout my time at the HCII she asked about who I was beyond my research, visited me when I was ill, and swapped stories with me. I am lucky to count Queenie as my friend. Thank you Queenie for supporting me as a student, but mostly, thank you for supporting me as a person.

The day to day challenges of this work could not have been overcome without such an amazing group of administrative staff. I am deeply indebted to Eric Davidson, Diana Rotondo, Ryan Reis, and Lindsay Olshenske for their help. I could not have found a quiet space to interview for faculty jobs, pay my participants, present this work at conferences, or collect data if it were not for them.

This work was informed by my conversations and collaborations with many students. I am grateful for the opportunities to work with Rajitha Pulivarthy, Nila Banerjee, Eileen You, Jackie Zhang, Ashley Hong, Nanxi Wang, Jasmine Shen, and Jade Hampton as this research matured.

Lastly, I am grateful to my family, friends, and mentors who encouraged me to persevere and saw promise in me. First and foremost, I owe my mother an enormous thank you for so clearly seeing that I could get a PhD even before I submitted my first application to the program. While it is not a path that she would have picked herself, she never doubted that I would succeed. I also am incredibly grateful to my newly minted husband, Chinmay Kulkarni, for all the support including taking care of Sinclair while I was on internship, pointing out goofy creatures while hiking, and worrying over how we would move to our next jobs so that I could focus on completing this dissertation. I am incredibly grateful to Dean Marlin for being a true friend in the midst of my PhD. He reminded me to take time to play by sailing regularly, helped me see how the PhD was an active choice that I needed to make daily, waited in airport baggage as I e-mailed this document to my thesis committee, and even helped me figure out the logistics of my Hindu/Agnostic wedding ceremony. I wouldn't be where I am today without him. I am also grateful to Rachel Vandenbrink for the hikes, the laughs, and the escapades.

There are countless others who I am thankful to for companionship and support including my PhD Cohort, the OH! lab, the Dev lab, the HCII PhD students, and the Pittsburgh Yoga, Sailing, and Rowing communities.

Table of Contents

Abstract **ii**

Dedication **iii**

Acknowledgements **iv**

1 Introduction **1**

2 Related Work **5**

2.1 What is a smart home? Situating IoT 6
2.2 IoT Customization for the Home's Idiosyncracies 8
2.3 A Brief Overview of End User Programming 11
2.4 Architectures and Toolkits for End User Programming of the Smart Home 15
2.5 Customizing Paper User Interfaces 21
2.6 Sustainable Socio-material Practices 23
2.7 Cultural Forms of Everyday objects 25

3 The Upcycled Home: Removing Barriers to Lightweight Modification of Everyday Objects **28**

3.1 Introduction 28
3.2 Methods 29
3.3 Participants 30
3.4 Procedure 30
3.5 Data and Analysis 31
3.6 Findings 33
3.6.1 Demographics 33
3.6.2 Division of Home Labor 33
3.6.3 Negotiating Social Boundaries through Ownership 37
3.6.4 Modifying Objects to Create the Aspirational Home 41
3.7 Discussion and Limitations 45
3.8 Conclusion 47

4 Understanding Family Collaboration around Lightweight Modification of the Home **49**

4.1 Introduction 49
4.2 Methods 50
4.3 Findings 52
4.3.1 Idea Generation 52
4.3.2 Contesting and Reconciling the Model of Home 56
4.3.3 Planning and Programming the Home 58
4.3.4 Potential User Burden 61
4.4 Discussion 63
4.5 Limitations 65

4.6 Conclusion . . . 65

5 The Lightweight Modification Framework for Augmenting Everyday Objects **66**
5.1 Object Relationships . . . 66
5.2 Family Relationships . . . 67
5.3 Societal Relationships . . . 67
5.4 Time Use for Evolving Relationships . . . 67
5.5 Employing the Framework . . . 69
5.6 Summary . . . 70

6 The IoT Codex: a Book of Programmable Stickers for Authoring and Composing Embedded Computing Applications **71**
6.1 Introduction . . . 71
6.2 Design Rationale for IoT Stickers . . . 73
6.2.1 Sticker Approach . . . 73
6.2.2 Participatory and Iterative Prototyping of IoT Stickers . . . 73
6.3 Sticker Fabrication and Operation . . . 78
6.4 IoT Codex Architecture . . . 80
6.5 Sticker Types and Use Cases . . . 83
6.6 Customization . . . 88
6.7 Design Space Validation . . . 90
6.7.1 Method . . . 90
6.7.2 Findings . . . 91
6.8 Discussion . . . 93
6.9 Conclusion . . . 95

7 Contributions and Open Questions **96**
7.1 Spatial End User Programming . . . 96
7.2 Adapting to Everyday Objects . . . 97
7.3 Tangible End User Programming . . . 97
7.4 Conclusion . . . 97

A Appendices **99**
A.1 Spatial Management and Floor Plans . . . 99

B Bibliography **100**

List of Figures

1 Two influential characterizations of programming systems from the early nineties as research on EUP began to take off. They graphically characterize how interactivity and mastery enable end users to customize computing systems. 13
2 Distribution of Object Modifications by Room 53
3 Results of User Burden Survey 61
4 Diagram of Lightweight Modification Framework 68
5 Photo Series of The IoT Codex in Use 71
6 IoT Sticker Buttons were redesigned with pre-programmed patterns that were grounded in everyday household objects. 74
7 IoT Stickers were redesigned to have a more compact footprint for *in situ* use. 74
8 Pictured are versions of two stickers from Prototype 1 of IoT Stickers and an RFID reader attached to a smartphone to serve as a mobile hub. 77
9 Stickers were redesigned to support tangible composition. 77
10 Priciples of IoT Sticker Interaction 79
11 The Button Sticker 80
12 The IoT Codex Architecture 82
13 The Toggle Sticker 85
14 The List Sticker 86
15 The Dial Sticker 86
16 The Wrapper Sticker 88
17 The color blocked annotations of the floor plan shows how family members spatially manage family norms. 99

List of Tables

1 6 Layers of Household Structural Change 7
2 Current design challenges for DIY-IoT [35]. 9
3 Reaction Handler's Architecture 16
4 Study Population Class Structure 32
5 Study Population Racial and Household Structure 34
6 Division of Labor Typologies 38

1 Introduction

> Every act of building betrays the environment, as it requires the displacement of ‘natural’ relationships.
>
> Richard Ingersoll, *The Ecology Question in Architecture*

Computers are escaping the desktop. Within the last century, computers exited research labs and made their way into the world. Once requiring the space of an entire building or room, they can now fit into a pocket and are likely to be swallowable as a pill in the near future [200]. In concert with this miniaturization, computers are making their way into everyday objects as part of the Internet of Things. For example, they are in the phones carried around in our pockets, embedded in our watches worn on our wrists, controlling a car's trajectory, and regulating indoor temperatures through heating and air conditioning. Likewise, interaction with computers has evolved from using Turing complete programming languages to model the atmosphere and compose music to fabricating personalized tools through Do-It-Yourself manufacturing (DIY Fabrication). Computers are not only entering the everyday environment, but actively making it.

Many enterprising computer programmers are helping to embed computers into the everyday environment. Early visions of the smart home focused on the architecture and services that serve as the infrastructure for a house and outfitted it to make it a 'smart' environment. To make a home aware of its inhabitants, researchers integrated electronic hardware like sensors with the house's physical infrastructure so that digital objects could be aware of their physical counterparts to better adapt to their contexts of use [116, 67, 79]. To realize this vision in commercial cases, homeowners would consult skilled contractors, network engineers, and systems administrators on how to accomplish this integration [79, 67, 151].

However, these heavyweight approaches are ill suited to the reality of home life. They are unable to augment the idiosyncratic arrangements and "clutter" families use to organize and give meaning to home life [211, 232]. Clutter helps track the state of projects, coordinates hand off to others, and externalizes the family's mental model of home. The ways that families use their artifacts resists generalization and differ from family to family.

A more lightweight approach introduces IoT to the home in a piecemeal fashion through replacing consumer goods. For example, current, commercially available IoT devices support families in upgrading their homes by switching to internet connected versions of common household objects. This includes typical American consumables like light bulbs [6], audio speakers [4], and blinds [5]. To adopt smart home infrastructure, this approach asks families to replace their existing possessions with new internet-enhanced versions. IoT services can even automate replacement. Disposable goods like laundry detergent can be replaced at the click of a button so that even the most ephemeral of household possessions can be part of IoT [52].

However, by asking households to replace their possessions in order to adopt IoT, IoT is accelerating household waste. From a material point of view, this replacement directly displaces household items and redirects them to the household's waste stream. If these items are electronic and not treated with care, this can be especially damaging to the environment [117, 174]. Yet, this waste also includes the nuanced relationships people have to their possessions, and so, erodes household meanings and ways of life. Discarding possessions to make room for new ones means owners must let go of their attachments to these objects and eliminate the entrenched routines and practices sustained by them [9, 163]. When artifacts newly enter the home, owners adapt them to their context in an expression and reach of their agency [9]. Through the adaptation process, possessions acquire unique meaning that reflects the owner's identity [91]. New technologies not only enable, but also constrain what routines, meanings, and identities can possibly be expressed [90]. In so doing, they shape the interests and identities of their owners [90]. Accelerating—and ultimately automating replacement—of household items substitutes device agency for individual agency to reshape household members' relationships [70]. Owners devote time, attention, and energy to their possessions as a form of care and maintenance so that they may last [23]. Before the advent of personal computers, ethno-archaeological studies found that over one third of household objects entered the home in used condition [194]. More recent work on product lifecycles uncovered the importance of object attachment, appropriation, reuse, and lateral cycling for how people treat their possessions in the long term [163, 174, 221]. This investment in household items is part of enacting and cultivating one's values: lights that automatically turn off erode the ability of parents to teach their children to not be wasteful [90]. Fundamentally, there is a tight relationship between what a household discards and its values. By asking owners to replace their possessions, IoT adoption culls the meanings and ways of life that have accrued in a household over time and can potentially erode the agency, values, and identities of families.

This raises the question of whether IoT should require replacement. Scholars have begun to consider alternative frameworks wherein computing is introduced. For example, computing could be an enabler of circular economies [30, 148], DIY manufacturing and open source hardware [173, 26, 34, 96], peer-to-peer exchange systems [231, 64, 71], and frugal scientific innovation [229, 53]. These research directions emphasize alternative modes of fabrication and adoption that rethink replacement and the decision-making discretion afforded to computing's end-users. Thus, IoT could be part of a circular material flow to support artifact renewal. If end users had control over how IoT is embedded in and customized to pre-existing artifacts, they could decide which pre-existing meanings and ways of life are discarded or sustained through IoT's adoption.

So, who should have this control over IoT? As a home technology, IoT is traditionally portrayed as a luxury good associated with discretionary spending (see [87] for the role of the discretionary user). However, discretionary decision-making does not adequately characterize householder choice. Structural factors limit families' power and autonomy over integrating systems like IoT into their household. For example, home ownership [63] or internet access [31, 236] constrain whether

IoT adoption is even a choice available to families. Societal structure also shapes household social dynamics around who is able to transfer their professional skills to the use of home technology, who occupies a household role that requires multi-tasking and fragments attention, or who has the leisure time to focus on investing in new technologies [187]. Structural factors, like socio-economic class, limit both capability and discretion to adopt IoT technologies.

Instead of creating a discretionary good, we could develop IoT's innovation under constraints of frugality. This could open up its downstream adoption to a wider demographic. For example, the project Foldscope developed a paper-based microscope for <$1 [53]. This made microscopic technology accessible to a global audience. If IoT were cheap, it could likewise reach a global audience. If an IoT artifact were to cost $<$ 10 cents, it would have the potential to reach a much larger demographic.

To begin to realize an IoT where anyone-regardless of their background-could give computational behavior to their everyday objects and shape that behavior, we need an IoT that is lightweight, cheap, and customizable. This dissertation examines whether it is possible to do this by focusing on whether the adoption of new computing, such as IoT, could be achieved as a process of upcycling. This dissertation adopts the term *upcycling* for its approach to a lightweight, cheap, and customizable IoT because it attends to IoT's material implications by prolonging artifact lifecycles and leveraging the existing routines sustained by them. Instead of replacing domestic possessions with internet-enabled equivalents, upcycling could support their renewal by retrofitting them with the latest computing capabilities. On this Upcycled IoT approach, upgrading to a smart home consists of augmenting domestic possessions with an internet connection and related IoT services. This dissertation centers end user discretion and decision making within the upcycling process by conceiving of lightweight upcycling as a form of end-user programming in response to the above challenges for domestic IoT. End user programming would enable end user to adapt IoT's user interface to accommodate existing possessions. To enable this kind of IoT, this work uses emerging techniques in backscatter, wireless communication to create a battery-free and wireless user interface that could incorporate existing possessions. Consisting in interactive stickers, this user interface enables everyday objects to be upcycled with internet capabilities through tangible linking. By being grounded in the home's already existing networks of relationships, values, and routines, these upcycled objects could offer families greater control over upgrade costs and discretion over what household norms and legacies might be dismantled in the process of introducing IoT.

To demonstrate the possibility of an Upcycled IoT, this work shows:

- Ways that socio-material practices are implicated by and could be reconfigured when domestic possessions are augmented by IoT
- Techniques enabling existing possessions to be transformed with IoT services
- A system to enable end-users to customize IoT to suit their domestic context

Taken together, these contributions show how an upcycled IoT could support installing, customizing, and managing the introduction of new IoT capabilities into the home. As a use case, these contributions to open problems in domestic IoT show how the challenges of computing's adoption could serve as sites for end users to have tangible control over how IoT is integrated in their home.

This dissertation first reviews the related literature on end-user programming, IoT, and socio-material practices as they relate to the home. As will become apparent, this latter topic quickly engages research questions on sustainability and this related literature will be covered to the extent that it is relevant. The rest of the dissertation is organized according to the stage of research and how far it advances this disseration's questions. First, chapter 3 and 4 cover formative research on household socio-material practices and the ways in which collaboration and coordination are marshalled to support adopting new technologies into the home. Then, chapter 5 introduces the resulting Lightweight Modification Framework and design guidelines for an upcycled IoT system. Chapter 6 describes the system and tangible interface—called *The IoT Codex*—built in response to the formative study's guidelines and resulting framework. It also characterizes the design space for an Upcycled IoT as enabled by the system and assesses it with end users. Finally, chapter 7 ends by synthesizing this dissertations findings and reflecting on its limitations. The conclusion also outlines open questions and future work.

2 Related Work

> It is exactly as though the physical items had been gathered together from widely separated sources and bound together to form a new book.
>
> Vannevar Bush, *As We May Think*

As a technology that is perpetually on the horizon, the smart home has been the focus of research for decades. Scholarship in this area has largely focused on achieving Marc Weiser's vision of ubiquitous computing in the everyday life of the modern home [225]. Instead, this dissertation adopts a vision that goes back earlier, to Vannevar Bush: a vision of computing's reach into everyday objects as a new kind of book [39]. Like the ubiquitous computing vision, everyday interactions with objects could benefit from computing's capabilities. Yet when envisioning how new computing capabilities could have lasting benefits for society, Bush needed to create an analogy with a book and scholarly practice to convey the possibilities computing afforded. His vision for computing grounds its benefits in a tangible, familiar form that leverages socio-material practices so that computing might augment these interactions rather than replace them. Bush's vision was later borne out as computing evolved. One of the 20th century's computing pioneers, Alan Kay observed that the concept of computing literacy arose from the the belief that the computer would evolve to be more like a book [126]. The central difference between Weiser and Bush's visions for the future of computing has to do with what it means to be literate within the vision of computing they put forth, and ultimately, the effort a person would need to put into using computing technologies.

Ultimately, this dissertation will propose a smart home system that takes the form of a book so that families can engage and acquire the skills necessary to have meaningful control over this system. To advance this proposal, this chapter draws upon decades of research on the smart home to make the case for leveraging existing socio-material practices and familiar forms when designing and developing cutting-edge computing technology for the home. This approach can not only offer sustainability benefits, but also exciting new avenues of research and approaches that center end-user creativity and discretionary decision-making. To make this case, this chapter provides a cursory overview of smart-home research to draw out lessons that households are clutter-filled places with idiosyncratic arrangements. To adequately design for this space, systems need to support customization instead of taking a one-size-fits-all approach. Thus, this chapter next reviews literature on tailoring systems and end user programming (EUP) of IoT so that households might customize computational services to suit their context. Next, this chapter reviews the technical infrastructure needed to support this approach by examining systems, architectures, and toolkits created to undergird these EUP approaches. While these systems might enable an appropriate approach, a focus on these systems alone would be remiss without in turn developing an understanding of the socio-material practices and cultural backdrop that serve as domestic context for these systems which could be undermined by these tools' approaches. Thus, this chapter draws on the relevant body of work on socio-material practices in the home to identify the limitations and opportunities that this dissertation will seek to address. Finally, this chapter makes the case for an interactive book for domestic IoT by reviewing research on paper-based computing and interactive books to show how extending this line of research to address open questions of end user programming for IoT in the home offers opportunities to address some of the challenges raised by research on the smart home.

2.1 What is a smart home? Situating IoT

Smart home technologies inherit from a ubiquitous computing vision. This vision seeks to extend computing beyond the desktop to the everyday objects that make up our physical world: like desks, offices, other people, the weather, trees, and even chance encounters [225]. This agenda has inspired researchers to seamlessly integrate new technologies into the environment so that technology might fade into the background to leverage a phenomenon of psychology whereby successful technologies are so woven into the fabric of everyday life that they are indistinguishable from it [225]. By envisioning ways in which computing becomes woven into the fabric of home life—from an alarm clock starting coffee at the owner's request to a smart pen capturing quotes from a paper newspaper and transmitting them to the office—seamlessness purported to free up a user's time and attention to focus on goals beyond using the technology itself.

In the early days of ubiquitous computing, researchers predicted that embedding computing in the home would have large social implications. They speculated that these implications included altering household routines, changing how household members signal control of living space, and even social conventions themselves, such as standards of household labor and good parenting [67]. If the television or game consoles offer precursors for smart home technologies, there would be negative impacts on family members' attention and relationships. Family members found their relationships with other members strained by the attention these devices commanded, and frequently found themselves ignored in favor of these devices [2, 10]. In contrast, when these devices support shared usage, household members view these devices as fostering quality time together [10]. Thus, what social context a smart home technology design assumes from the outset can drastically shape a household's social dynamics.

Early research envisioned the smart home would be a smart building. To make a home aware of its inhabitants, researchers integrated electronic infrastructure like sensors with the house's architecture so that digital objects could be aware of their physical counterparts to better adapt to their contexts of use [116, 67, 79]. Skilled contractors, network engineers, and systems administrators would be brought in to consult with homeowners on how to accomplish this integration [79, 67, 151]. Scholars drew from architectural research to develop a model of the home consisting of 6 layers of structural change [27, 49, 188]: site, structure, skin, services, space plan, and stuff (See Table 1). These architectural understandings helped with

Layers of Household Structural Change

SITE (Fixed)	This is the geographical setting, location, and the legally defined lot, whose boundaries and context outlast generations of ephemeral buildings.
STRUCTURE (30-300 yrs)	The foundation and load-bearing elements are perilous and expensive to change, so people don't. These are the building. Structural life ranges from 30 to 300 years.
SKIN (20-30 yrs)	Exterior surfaces now change every 20 years or so, to keep up with fashion, technology, or for repair.
SERVICES (20-30 yrs)	These are the working guts of a building: communications wiring, electrical wiring, and plumbing. Buildings are demolished early if their outdated systems are too embedded to replace easily.
SPACE PLAN (3-30 yrs)	The interior layout – where walls, ceilings, floors, and doors go. Turbulent spaces can change every 3 years or so; exceptionally quiet homes might wait 20-30 years.
STUFF (Continual)	Chairs, desks, phones, pictures, kitchen appliances, lamps, hairbrushes; all the things that twitch around daily to monthly. Furniture is called *mobilia* in Italian for good reason.

Table 1: Reproduced from [188], the table shows 6 layers of material change and their time span according to architectural research.

designing for the time span integrating computing at these layers of the building's construction would presuppose.

At the same time, this architectural understanding revealed two different trajectories within the smart home research community. Drawing on the architectural layers framework, scholars argued that context sensing, digital technologies, and digital infrastructure engaged with the services layer [188]. However, most empirical research with householders focused on the layers of space plan and stuff [188]. This empirical research promoted several categories for durable goods to make a home smart (*e.g.,* information appliances, interactive household objects, and augmented furniture) [188]. Yet, this framework highlighted how these empirically motivated design proposals were disconnected from the technical advances that were being developed to enable smart homes. As a result, scholars called for empirical research to understand the broader community of stakeholders involved with the services layer [188]. Thus early, enabling technology for the smart home contributed heavily to the service layer, but empirical research was largely silent on the social implications of introducing computing at this layer.

Subsequent research investigated wireless networking and home renovations to begin to accumulate empirical understanding of the service layer of smart homes. In this literature, a central conflict was uncovered: the family's mental model of home often does not align with smart home networks. As a result, family members struggled to install and configure home networks [67, 86]. The researchers concluded that they needed specialized knowledge approaching that of a systems administrator [67, 86]. In what researchers termed a *control paradox*, families created complex conventions to remind themselves of the network's configuration to gain control. Yet, they increasingly felt their control erode as the complexity of the network grew, and household members began to specialize in managing it or making upgrades [49, 91].

In addition to the Control Paradox, researchers found that households adopt divisions of labor that can negatively interact with the introduction of IoT in the home. Research on home renovations found that specific household members specialize in learning and programming IoT devices [151, 232, 35]. This verifies earlier findings that households have difficulty sharing access and control of devices among family members [49, 189]. Two trends contribute to this difficulty: specialization gives rise to skill gaps, and household artifacts can be gender stereotyped [49, 189, 232]. When one household member programs a device and another must live with it, family members' emotional reactions to those programs have implications for acceptance [47, 35]. This later stage research found that smart home infrastructure focused at the service layer was poised to exacerbate social disparities and erode inhabitants' autonomy and control over their home.

2.2 IoT Customization for the Home's Idiosyncracies

Families do substantial work to integrate new computing technologies within the household's social dynamics in order to make them acceptable. Upgrading to a smart home involves making critical decisions about which devices enter the home and how they are configured for the household's members. Households are unlikely to adopt smart home devices as a wholesale upgrade [67]. Instead, these devices enter the household in piecemeal fashion as small improvements or reappropriations [49, 67, 167, 35]. This leads to a highly complex network of home technology that will be unlikely to conform to any single manufacturer, domain/platform, or even technical standard [49, 67]. Families work to weave IoT into their routines, and this process is critical for making these devices a success [213, 232]. The adoption process is disruptive to the home, and family members will resist new devices becoming integrated if they perceive the process as too demanding [91]. Smart home adoption processes that focus on piecemeal integration are likely to be better suited to the household ecosystem than processes that require wholesale overhaul because they can better support the work families to do to integrate new computing technologies.

To provide this support, piecemeal adoption needs to fit with the idiosyncratic and situated practices households develop for running home life. These practices are unlikely to generalize to other households or be adequately supported by commercial IoT systems. In an attempt to distill predictable tasks that could be computationally supported from daily routine, such as making coffee [225], researchers found that home life resisted routinization [56]. Instead, the breakdowns, exceptions, and improvisation that characterize daily life need to be supported [56, 137]. Re-

Areas in DIY-IoT that Need to be Supported

REFLECTION	Reflecting on routines to identify appropriate IoT interventions
DEBUGGING	Debugging in situ to help with the test cases that occur as part of daily routines
EVALUATION	Family feedback on IoT intervention including emotional reactions
INTELLIGIBILITY	Interpreting a device's state, which components are active, and the structure of the system
INSTALLATION	Easy ways to install IoT through guidance on material interference and component form factor such as attachable/detachable sensors and actuators

Table 2: Current design challenges for DIY-IoT [35].

searchers argue that to make a home smart, systems should thoughtfully create a role for human-in-the-loop intelligence [213]. This argument undermines calls for AI-complete systems to fully recognize home activity. Instead, scholars have called for object-centered practices that attend to care, maintenance, and repair as highlighting a missing understanding for the ways in which values are enacted in householders' relationships to their objects and the creative processes in which home life is situated [83, 60, 46, 115, 214]. Researchers argue that if the smart home were to leverage households' organizational systems, they would need to make space for 'clutter' and idiosyncratic systems in which members create place and give meaning to their home [211]. They caution against ubiquitous, centralized systems: *"Having information stored in one central place, displayed throughout the home, and smartly following us from room to room begins to disassemble the choices we have made in where to put things."* [211]. Findings from smart home research continue to caution against systems that would be recognizably the same across households, and instead, call for systems that support the ways that households use inventive material practices and idiosyncratic arrangements to curate their home. These situated household norms form the backdrop for piecemeal adoption.

When designing IoT for the adoption process, a central question arises over how to support end user agency and control over the system. The control paradox shows how home networking systems face persistent challenges in supporting end users in installing and configuring wireless services. However, automating these processes confronts its own set of challenges. When Amazon began distributing Dash buttons to automate home purchasing decisions through a wireless button, public opinion balked at the envisioned future the button promoted [52]. The Dash button future presented *"the dream of domestic life as a perfectly calibrated, automated system"* [52]. When the embedded system seamlessly fades into the background, the moment to reflect on one's choices and do things differently is missing and this can effectively erode one's own agency [52, 190]. This problem of eroding agency through seamlessness is especially prominent in the Dash button's design: its purpose to automate household choices and in particular, purchasing choices.

Charting an alternative vision for ubiquitous computing than the seamlessness interaction proposed by Weiser, *Engaged Computing* encourages researchers to consider how embedded systems could extend human abilities to be constructive, creative, and ultimately, in control of the actions they take in the world [190]. Empirical findings show a persistent need for supporting discretion and control over how the introduction of computing into domestic life shapes social roles and co-constructs home dwellers identities [232, 108]. To address this need, Do-It-Yourself IoT (DIY-IoT) could offer end users this discretion and control over how smart home technologies are woven into routines and relationships [9, 42, 35]. Research suggests that a DIY IoT is likely to support greater self-expression [58]. In an analysis of three case studies, researchers argue that IoT can support the owner's identity, agency, and autonomy by giving control to owners over their participation, sharing, and arrangement of devices within the home [9]. Moreover, this approach can foster creative collaboration in families when members are empowered to re-purpose and build off each other's ideas using nearby objects [58]. However, to date, DIY-IoT needs to be designed to support 1) reflecting on routines to identify appropriate IoT interventions, 2) debugging, 3) facilitating family feedback on IoT intervention, 4) interpreting a device's state, and 5) installation [35]. How these DIY-IoT processes interact with other activities and values in the home will need to be surfaced to make progress on this agenda.

DIY-IoT is entangled with nuanced social dynamics related to *what it means to program the home* and *who should be responsible* [13, 189, 35]. Early work on software tailoring culture developed a model of inequities in who is positioned to tailor computational behavior: greater customization will require greater programming skills [144]. However, a community can foster a climate in which more expert members assist lower skilled members with learning the things they need to gain greater control of software systems [76, 144]. Despite work to nurture community dynamics for skill development, barriers remain. Empirical research on EUP for domestic IoT continues to see programming skill and access concentrated in one family member [35, 232, 122]. Family members frequently adopt gender-stereotypical roles with respect to who builds the home network and who learns to program the device [151, 189, 35]. Design will need to facilitate community relationships that foster tailoring IoT to counteract structural forces that push towards unequal concentration of tailoring skills in one family member (especially, if this concentration is stereotyped).

Many of the community challenges facing DIY-IoT in the home are not unique to it (though *some* clearly are, as listed above in the design needs for DIY-IoT), and are faced by end user programming systems in general. Early work in this area highlights the benefits and limitations when systems try to address the challenges for tailoring communities in system design. A prototype language enabled more skilled users to write code for the actions of on-screen buttons and novices to copy and reuse those buttons without needing to know anything about the underlying infrastructure [144]. Other systems created metaphors—such as jigsaw puzzle pieces or magnetic poetry—for novices that abstract away from the system's details, but can be combined and arranged to compose small programs [102, 216]. This allows for users to have a direct role in creating the resulting capabilities, and shows how control constructs, component design, and appropriate metaphors can lower barriers in tailoring systems for novices. Even so, Blackwell points out that shortcomings in a system's usability can be mitigated by social adaptation [21]. Some of the best research on this topic comes from Mackay's early study characterizing community customization dynamics in a workplace setting and how it evolved over time [141]. Yet, the social processes involved have been relatively neglected in the research on EUP [21]. This is especially true for the home where community challenges for EUP systems remain understudied.

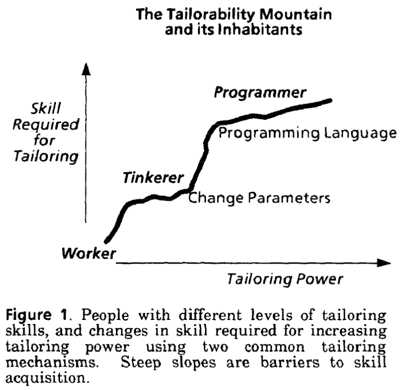
2.3 A Brief Overview of End User Programming

End user programming has a long and prolific history. This dissertation will not attempt to cover EUP's history in its entirety. Instead, this section provides an overview of prominent conceptions of what EUP is, who end users are, what it means to program as an end user, and how the design of user interface components are a central concern of EUP research. In covering these topics, this section will claim that an EUP system is an interactive system that supports end users with expressing their ideas to a computer and leveraging computational support to work out problems with their ideas. In so doing, EUP preserves agency over a system by deferring to the end user what ideas and problems are of interest as well as how a reasonable solution will take shape.

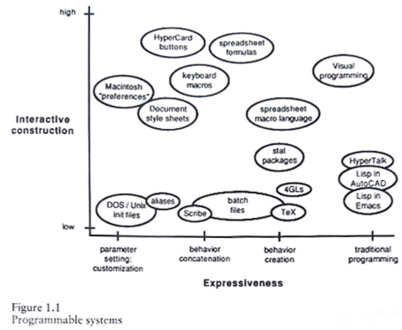
As the field of computer science emerged, so too did end user programming. It co-evolved alongside the rise of programming languages and user interfaces. In doing so, end user programming encapsulates both being able to use a computer as well as being able to directly shape what a computer can do. Alan Kay characterized end user programming of a computer through analogy: as a combination of both using and shaping a tool [126]. He further likened this dual characterization to literacy where being literate includes being both able to read and to write [126]. Central to Kay's characterization is a conception of end user programming as interactive. Drawing on Licklider's *man-computer symbiosis*, Kay highlighted how interactive systems like Ivan Sutherland's *Sketchpad* demonstrate how computing can support this dual role by helping end users formulate technical problems that the computer can provide helpful feedback on, if not outright solve [140]. The user interface became a tool to help with articulating and reflecting on a problem. In essence, interactive systems support thinking through a problem. *JOSS* introduced a dialogue-like interaction between the programmer and the computer through the programming language itself that printed out onto 8x11 paper as a extension of a notebook [201]. To prompt the user to reflect and read back through their input, the computer would respond with *"Eh?"* when the programmer made an error such as a malformed expression [201]. Building on this notion of computer literacy, *Dynabook* codified the belief that computing would be more like a book than a swiss army knife [126]. End user programming wasn't simply "about getting and conveying information, but the very act of learning and doing them expands one's horizons and adds new ways of thinking about the world" [126]. Thus, the design of end user programming needed to account for pedagogical benefit by expanding what a person is capable of, not merely automating it.

Partly due to end user programming's co-evolution with user interfaces and programming languages, definitions of end user programming abound. They also diverge in notable ways. Kelleher and Pausch define programming *"as the act of assembling a set of symbols representing computational actions"* [114]. Their definition occurs within the context of creating a taxonomy of programming environments and languages for novice programmers. So their definition is not a definition of end user programming per se. At the same time, their definition is designed to exclude programming by demonstration (PBD), a prominent sub-branch of end user programming. Kelleher and Pausch argue that PBD does not allow the end user to learn to program since there is not a way to accurately predicting what program will be produced [114]. Their definition emphasizes pedagogical benefit by focusing on symbolic composition, and excludes some kinds of end user programming on the basis of it. Ko, et al define a program as *"a collection of specifications that may take variable inputs, and that can be executed (or interpreted) by a device with computational capabilities"* [120]. Their definition is silent on whether composition or learning is important. Further, the way in which the specifications are collected is left open. To define end user programming they follow Nardi: an end user is simply someone who uses a computer, and end user programming is programming done to achieve a result of a program for personal use [120]. For her part, Nardi explicitly excludes learning to program, but she does not rule out symbolic composition. Nardi excludes learning because she believe that this conception rests on a fundamental mistake: that end users want to turn their task into a programming problem. To emphasize how this view might be mistaken, Nardi explains, *End users are not 'casual,' 'novice,' or 'naive' users; they are people such as chemists, librarians, teachers, architects, and accountants, who have computational needs and want to make serious use of computers, but who are not interested in becoming professional programmers"* [157]. Blackwell echoes this criticism, and elaborates that conflating end users with novices or students introduces problems with generalizing from research results to end user developers who are skilled at their own work and have likely completed formal education [21]. Keeping with Nardi and Blackwell, we will call the mistake of conflating end users with novices the *Novice Fallacy*. The novice fallacy does not eliminate the need for a learning or pedagical benefit in end user programming, but it does caution against creating end user programming languages designed to teach people to become professional programmers since this larger goal is likely counterproductive for end users.

Definitions of end user programming quickly begin to encompass the end user's perspective. Unlike many of the scholars above, Nardi essentially includes a person's goals in her definition of programming: *"the objective of programming is to create an application that serves some function for the user"*. Nardi argues that programming's definition ought to include the objectives a person has in order to avoid confining the definition to any particular technology or language. From her point of view, writing high level declarative specifications or creating diagrammatic representations can be treated equivalently as programming because they involve the same basic



(a) Tailoring skills as a function of tailoring power from [144].



(b) Interactive construction as a function of expressiveness from [157].

Figure 1: Two influential characterizations of programming systems from the early nineties as research on EUP began to take off. They graphically characterize how interactivity and mastery enable end users to customize computing systems.

activities and skills even if one takes less time. What the end user's thought process is and how the user interface supports this underpins many of the open questions for end user programming. A number of scholars focus on the user's intention. Research on end user software engineering becomes relevant when the end user shifts their intention from creating a program to serve their individual needs to sharing it with others (even if sharing the program was not initially intended at the outset) [120]. This emphasis on user intention is central to programming by demonstration research because of its critical task of inferring what parts of a user demonstration should be generalized [126]. Even for EUP research unconcerned with tracking a user's intentions at the stage of design or demonstration, understanding why a person is engaging programming at all has large implications for what the EUP language should support [114]. Kelleher and Pausch divide the user's motivation into two different categories: a) learning programming for its own sake, or b) using programming in support of another goal [114]. Even within Kelleher and Pausch's categories, the subcategory *mechanics of programming* constitutes the substantial concern for the developed systems even if they approach the topic from differing directions [114]. Perhaps because programming mechanics have dominated concerns for system design, the psychology issues at stake when the user interacts with an end user programming language have driven a number of research questions on how to improve the usability of end user programming languages [21]. Whether end user programming is defined in a way that explicitly articulates a role for the users goals, researchers agree that understanding end user perspectives on end user programming languages is essential for determining how well the end user language is designed.

So what aspects of the end user perspective should EUP be concerned with? First, whether the end user is learning to program or simply learning to use programming in support of other goals, there is a continuum of customization capabilities that will partly determine how successful they are. Some scholars have characterized this continuum as the threshold and ceiling of a system: *"the threshold is how difficult it is to learn to use the system, and the ceiling is how much can be done using the system"* [153]. An ideal system has a low threshold and a high ceiling. Early scholars thought that in order to progress along this continuum, the end user also needed to similarly advance their tailoring skills (see Figure 1a) [144]. As pointed out by Nardi earlier, this pits highly skilled workers at one end of the spectrum and programmers at the other. Another way to think of this continuum is to focus on what is accomplished and how. Thus, changing parameters of a program lies on one end of the spectrum and programming languages on the other (compare how Figures 1a and 1b both share this intuition). Nardi calls this continuum *Expressiveness*. In keeping with both Nardi and Kelleher's points that advancing along this continuum shouldn't necessitate becoming a programmer, we add the need for *"wide walls"*. If a system has wide walls, it enables a person to explore many different kinds of projects [185, 186]. This focuses system design on nurturing the person's own interests and passions [185]. By designing for wide walls, an end user programming language supports creativity by *"define[ing] a space to explore, not a collection of specific activities"* [185]. Following the above consensus, we argue that an end user programming system is expressive if it has a low threshold, high ceiling, and wide walls. Thus, an end user programming language should support the end user's expressivity.

In early developments of computer literacy concepts, *interactive* systems served to prompt reflection and facilitate thinking with computing tools so that end user programming languages could better support expressing a problem to the computer. As one of the first end user languages, JOSS, was developed to support smooth interaction for mathematicians using a computer to enable solving mathematical problems without having to learn to be a programmer [126, 217]. Programming by Demonstration, a prominent sub-branch of end user programming, minimizes the need to learn to program by employing user interaction with interface elements to specify what routines should be automated by the computer [126]. One of the PBD earliest systems, *Pygmalion*, created a visual language using *icons* to facilitate interaction between the user and the computer by editing pictures [126]. Pygmalion showed how concretizing program data—like working out equations for factorial statements—with pictorial elements that represents the system's state facilitated editing the program as an artifact [126]. An icon based interface was thought to be more user friendly because it supported analogical reasoning in contrast to symbolic reasoning [126]. Pygmalion's innovation was to ask the user to reason analogically using the interface's iconic language while the system generalizes from that interaction to other cases through automation. Pygmalion, and other early interactive systems, used dialogues to prompt reflection and facilitate refinement to an end user's expressed problem. Yet, these systems assumed that the end user would be demonstrating an action that they already knew how to do and merely needed PBD to automate repetitive tasks [126]. However, the most useful kind of end user programming is likely to be in contexts where the user doesn't know what they want to do, and the system will have to provide support for helping end users discover their needs [157]. Thus, the pedagogical benefit that end user programming needs to provide has more to do with interactivity helping the end user refine their ideas than it does with helping them learn to program.

The ability of icons to concretize data enabled substantial progress on end user programming's support for refining ideas. Concretization allowed for the end user to iterate on a program using a representation of the program that is both understandable to the computer and themselves. Concretization was designed to solve gaps between the programmer's mental model and what the computer would accept [126]. According to Smith, the gap arose because computers at the time only accepted Fregean representations and were unable to accommodate analogical representations [126, 203]. Analogical representations, such as maps, give *"information about the structure of what is represented"* [203]. In contrast, Fregean representations, like predicate calculus, give information about the relationship between functions and their arguments, but they do not require a globally consistent interpretation [203]. Analogical representations were thought to provide benefits over Fregean because they allowed for relationships in the represented context to be discovered by enforcing consistency [126]. For example, Pygmalian queries the end user when their sketched problem remains unresolved to nudge them to fill in remaining arguments similar to JOSS's prompt to reflect, *"Eh?"*. Concretization made enormous strides by *"us[ing] graphical means to express graphical concepts"* [153]. To characterize how consistency could be leveraged with analogical representations, Nardi coined the term *"visual formalism"* because the representation provided an orienting framework with which to organize data. When the visual formalism is editable, as with Pygmalian, the end user can adapt the system to their context.

2.4 Architectures and Toolkits for End User Programming of the Smart Home

Architectures for end user programming (EUP) and for the smart home have largely developed along two separate trajectories. Most end user programming architectures focus on flushing out Ivan Sutherland's vision of interactive computing by means of pointing at graphical representations to enable live sketching with computational support [210]. The rise of interactive computing generally drove the emergence of event-oriented architectures. Somewhat later, architectures for the smart home began to focus on supporting context-awareness and otherwise making a building aware of its inhabitants to fulfill the ubiquitous computing vision of the 1990s [196]. Architectures that began to merge the two capabilities—supporting end user programming and smart home interaction—emerged only within the last couple of decades as the rise of tangible and physical computing began to encourage researchers to focus on how to enable designers to rapidly prototype user interaction.

Control Component	Table-driven syntax analyzer; Tables contain ring-structures and include a model of the state diagram; Contains references to the routines, program blocks, and instructions for execution
Procedure Component	Routines, program blocks, and instructions for execution
Supervisor Component	Contains routines for handling interrupts and maintaining the display

Table 3: A summary of the principle components of the Reaction Handler.

The Emergence of Event-oriented Architectures

A central question for end user programming architectures—and for early interactive systems in general—consisted of managing interaction with the user. The architecture needs to provide for and manage feedback even during intermediate stages of interaction. For example, consider when a user makes a stroke with a pen of a screen based interface: the display needs to update digital inking for every location the pen has travelled throughout the duration of a stroke even when the pen has not been fully lifted from the display to cue the end of the stroke [160]. State machines became an intuitive way to formally reason about system feedback and represent when the system needed to track and store input data—such as pen location data—to provide system feedback (*e.g.,* the display's inked line) [160]. This natural correspondence between a formal, visual language and the way it represented how the system handled feedback to the user on its state enabled the development of a procedural language—the *Network Definition Language*—to leverage the correspondence between representing interaction with a state machine and the programming language to create the user interface component [160]. In a form of bilingual programming, the *Network Definition Language* complemented another, *control language*. These considerations informed the underlying architecture for this system— called the *Reaction Handler*: control component, procedure component, and supervisor (see Table 3) [160].

A decade later, Smalltalk introduced the Model-View-Controller (MVC) architecture that has persisted to the present day. MVC was designed to 1) support highly interactive software development, and 2) provide a general set of components for programmers to be able to easily create portable, graphical applications [124]. Like the Reaction Handler, MVC uses a 3-way factoring of a program's architecture [124]. A *view* presents information on a screen [82]. A *controller* represents the ways a user interacts with the screen view [82]. Finally, the *model* is the object that stores the information to be accessed by the *view* [82]. Smalltalk separated these components out into their own separate classes, and these classes could be extended [82]. This allows for user interface components to be defined and written once, and then shared so that user written applications could plug in these components to specially written domain-specific information [124]. MVC leverage object-oriented programming (OOP) to facilitate separation of concerns and allow for input devices to create messages to be passed throughout the system [124]. Notably, menus, like user interfaces, are often thought of as view-controller pairs [124, 82], but were categorized as controllers by MVC's architects because they are classed as an input device [124].

Advancing MVC, the Andrew Toolkit allowed user interface components to be composed together to form more complex components and allowed components to be embedded in one another [170]. Like MVC, the Andrew Toolkit separated views (the display) from the data (the information to be displayed, model) [170]. However, the function of the controller was distributed between the *interaction manager* (capable of making global decisions) and local decisions made by views [170]. This distribution of the controller's function allowed the Andrew Toolkit's composition capabilities. Views were represented in a view hierarchy with the utmost parent consisting in the interaction manager [170]. The interaction manager then translated input events (from keystrokes, mouse events, etc.) to the child views [170]. This occurred through message passing. Messages were passed along from the interaction manager to its child, and events were either handled by that child or passed along to its children until the event has been handled [170].

MVC and the Andrew Toolkit simplify programming user interface components for developers by using an OOP paradigm and message passing. Instead, MIKE exploited the OOP basis of user interface components to allow programmers to simply declare a user interface element and thus, create one [169]. MIKE did this by mapping user interface components' capabilities onto the programming language PASCAL to make interface generation more user friendly [169]. Using a command procedure metaphor, MIKE divided behavior of the interface between *actions* and *objects* similar to the bilingual division of the Reaction Handler [168]. It did so to allow design teams to work in parallel with one another and reason over the same user interface components without requiring human factors experts to have programming skills [168]. Programmers were to focus on defining what the command function is to do while the human factors experts were to define the external aspects of the user interface through the MIKE interface editor [168]. Unlike MVC and the Andrew Toolkit, MIKE's architecture considered the roles of the different designers in its architecture: a programmer would largely make use of existing user interface components to adapt them to a specific domain and a human factors expert would worry about the look and feel of the interface using these standard components. Thus, the architecture should support their work in parallel by using standard interface components to coordinate work done on what MVC calls the view and controller through a rough working user interface generated by the team through a set of semantic commands [168]. MIKE illustrated the advantages of the programming basis for user interfaces by illustrating its benefits using a menu component. Menu items could be added by declaring the name of the procedure to add, and specifying a set of parameters: the menu to attach the procedure to, the external procedure, and a command key [169]. Thus, the programmer tailored user interfaces using application-specific code largely by choosing the relevant procedures and specifying the appropriate parameters.

Like MIKE, Garnet supports the programmer in customizing user interface components through by parameterizing their behavior rather than through sub-classing (as in MVC) [154]. Unlike MIKE, Garnet handles abstracting away from input devices so that these low-level details are hidden away from the programmer using *Interactors* [154]. Following MVC's tripartite separation, Interactors serve as the controller, the view handles output graphics, and conventional Lisp code supplies the model [154]. Garnet further uses constraints to tie the 3 together [154]. In practice, MVC implementations led to tight coupling of the controller and view (as the Menu examples illustrate), and so changes in input devices frequently led to views having to be re-coded [154]. Garnet argues that parameterization of built-in types rather than sub-classing nudges the programmer to separate the concerns of MVC by removing discretion over what information should be apportioned to the controller [154]. Garnet implemented 6 interactors based off identified behavior patterns from graphical user interfaces (GUIs), but noted that new interactors would need to be created for dramatically different kinds of interaction such as gesture of optical-character recognition [154]. Interactors used a prototype basis for its programming language so that programmers could declare an instance of an interactor that was close enough to the desired behavior and then override any parameters as needed [25, 154]. Further, a programmer could customize the interactor by supplying parameters [154].

Architectures for End User Programming of Context-Aware Systems

The OOPs toolkit introduced a suite of mechanisms for handling ambiguity in the user's input, but context-aware systems were designed to supplement input with contextual information to support disambiguating processes. Context carries implicit information that could be leveraged to trigger changes in a system's response [198]. To simplify building context-aware applications and reasoning about context-shifts, the Context Toolkit to 1) hide the complexity of unconventional sensors, 2) manage the details of providing relevant events for applications, and 3) provide reusable building blocks for context-aware applications [193]. Termed, *context widgets*, these building blocks are made up of *generators, interpreters, and servers* [193]. Like interactors and controllers, context-widgets provide an abstraction for input devices. Yet unlike them, user input to these widgets is implicit and so does not afford direct control over inputs. Generators encapsulate one or more sensors and the software that acquires the raw data from the sensor (i.e. transforming an id report from a sensor to a name associated with that id) [193]. Interpreters supplement raw data to provide higher level abstractions (i.e., inferring a meeting is taking place between two people raw data shows activity of two people in a room) [193]. Servers collect, store, and interpret data from other widgets [193]. While both OOPs and the Context toolkit help with reasoning over noisy sensor data, the Context toolkit's components facilitate composition and supporting the development of more complex components. The purpose of context widgets is to separate how context is acquired from how it is used [61]. Explicitly supporting composition enables the Context toolkit's actions to likewise make use of more complex events.

This early work in context-aware architectures serves as the basis for trigger-action programming for end user development. Context-aware architectures focused on separating out the basis for an if-then conditional [196]. This is evident in the separation of concerns provided by context-widgets. Trigger-action programming uses this separation to facilitate composition of context with appropriate actions [62]. Using techniques developed from GUIs, iCap allowed the user to compose situations for the conditional's antecedent by dragging elements from a pre-defined library of context sensors to slots that facilitated Boolean composition [62, 171]. Similarly, the user could compose the actions that they wanted to trigger when the situation was detected [62]. Thus, iCap made context-aware programming available without having to write any code.

Today, EUP is the means by which users can shape and control the computational behavior of IoT devices in the home. EUP has become widely available in commercial systems through services like If This, Then That (IFTTT). These systems make EUP tools for home automation available to a larger audience at a lower price point than previously available [218, 219]. Prior work shows that the trigger-action programming model of systems like IFTTT make programming IoT devices available to non-experts [62, 171]. As with tailorable buttons, IFTTT recipes can be copied and changed by community members [219]. This approach uses GUIs to encapsulate control flow constructs like if-then rules within user friendly abstractions like form filling [62, 171]. As a result of its widespread availability and preliminary successes, trigger-action programming has received substantial attention from researchers creating EUP systems for domestic IoT (*e.g.,* [151, 80, 29]). Thus, trigger-action programming has become the de facto standard for EUP for domestic IoT.

Trigger-action programming still encounters a number of usability issues for end user programming of IoT. Choosing between event triggers (signalling change in a sensor reading at a point in time) and condition triggers (detecting states, context, or when criteria are met) introduces many problems [80, 101]. One study found that selection of triggers and actions had the lowest usability ratings of all measured dimensions [80]. Scholars have speculated that this may be due to if-then rules ambiguously covering both events and conditions [101]. They argue that the abstractions available to users do not make implicit distinctions of the system's syntax transparent and leads to many debugging issues [101]. Yet trigger-action programming's challenges may run deeper. In its initial design, researchers conducted a linguistic analysis of participants' programming ideas, and this methodology lead them to conclude that conditional statements using sentential logic were best suited for representing end users' ideas [62]. Form filling provided an interactive dialogue to prompt end users to supply arguments and visually separates the antecedent from the consequent [3, 62]. However, form filling lacks procedural flexibility and is best suited to contexts that are most similar to filling out a paper forms like filling out medical charts [157]. This challenge with capturing the procedural complexity of the home with appropriate EUP abstractions has been observed is an outstanding issue for domestic EUP designs [29]. Like all EUP systems, finding the representation best suited for the context in question is no small feat [157]. However, given the lessons of concretization, trigger-action programming may be fundamentally hampered by its initial methodological commitment to linguistic analysis that analyzed participants ideas according to its use of Boolean connectives. This choice of analysis may have made a Fregean representation inevitable.

End User Programming for Physical Computing

To make progress on EUP for IoT, greater consideration needs to be given to how home IoT embodies computational properties and makes computing physical. Phidgets introduced the notion of a physical widget to the research community [85]. Phidgets abstract and package input and output devices to support simultaneously designing the physical interface alongside the software interface [85]. The Phidgets architecture consists in the *physical device*, *Wire protocol*, *phidget manager*, *phidget-specific COM objects*, *phidget API*, *phidget-specific API*, and *ActiveX controls*. A central innovation of this architecture was to use the Wire protocol to coordinate device specific information with the visual representation of an ActiveX control through the phidget concept. This coordination allowed for subsequent event handling to respond to relatively uniform information: the phidget type, a phidget instance ID, and associated event information indicating device state [85]. The phidget manager, phidget-specific COM objects, phidget API, and phidget-specific API generally manage interpreting this information through COM objects on a Windows machine and expose this information at differing granularities that are more or less specific to the phidget device through an API.

As physical computing became more commonplace in the research community, end user programming toolkits like phidgets became more commonplace as researchers sought to simplify and streamline the architectures supporting them so that tangible interfaces could be easier to develop. The Calder toolkit offered a similar architecture to phidgets, but focused on smaller components and support for wireless communication so that designers could iterate on the form factor of the device in tandem with iterating on its function [135, 16]. Papier-Mache simplified the process of adding input devices by using an interactor-style abstraction for input devices to support computer vision, RFID, and barcodes for an event-oriented architecture [119]. dTools likewise focused on early stage design and protoyping of tangible interfaces by supporting an authoring environment that allowed for physical controls to be plugged into a custom microcontroller and a visual editor to associate states and transitions with the newly attached control [92].

These environments for rapidly prototyping physical form alongside computationally supported function introduced an era of researching support for sketching in hardware. These systems began to emphasize creativity above and beyond the hardware-software stack. Physical computing kits were developed for e-textiles to enable a range of users to create wearable computing applications [32]. Scholars began to experiment with ways computing materials could integrate more fluidly with creative practices like sketching and design craft [16, 135, 33, 180]. For example, the precision and computation afforded through procedural programming were used to augment the fluid and open-ended processes of manually drawing and generating a pattern [107, 106].

2.5 Customizing Paper User Interfaces

As research on end user programming began to focus on creativity and rapid iteration, scholars began to make rapid advances in developing paper user interfaces. In particular, sketching, paper prototyping, and the persistence of paper-based management in workplaces made paper-based computing become an emerging site of new interaction techniques. These, I will argue, need to be architecturally supported.

Kinetic Mechanisms for Programming

Kinetic mechanisms in the form of pop-up, paper engineering techniques show promise for supporting EUP tangible affordances. While visual language abstractions successfully use shape to introduce programming concepts like control flow and variables [184, 171, 216], kinetic mechanisms' potential for programming behavior remains underappreciated. Electronic Popables introduced paper engineering's development of kinetic mechanisms to user interface design [179]. The project integrated storytelling with material form through paper shapes and electronic circuitry. Paper Generators harvested kinetic energy generated through tapping, touching, rubbing, and rotating paper mechanisms to self-power the embedded electronics using the triboelectric effect [111]. PaperID used a volvelle (a rotational mechanism) with a rotating RFID antenna to selectively bridge with the integrated circuit (IC) to enable user control over reads [138]. Similarly, scholars investigated bistable mechanisms to manage widget state [239, 205]. Since state machines drive user interface widgets [160], a pop-up mechanism's bi-stability provides a tangible representation and mapping between widgets and the system's state. Scholars also combined the bi-stable mechanisms of traditional origami crease patterns with the triboelectric effect to enable deployable forms [44, 178]. Fine-grained sensing of paper shape change is unable to capture precise measures of bending and strain leading to introduction of a threshold at which the deformed mechanism transitions to a discrete state [228, 240, 43]. To date, paper-engineering techniques for user interfaces focus on sensing and shaping their affordances.

Insofar as this research focused on programming, it has been confined to shape change and predictable control over bi-stability. The early interest in paper-engineering emphasized biomemesis and modelling naturally developable forms with mathematically rigorous origami patterns that could be computationally simulated [84, 128, 121, 220]. HCI research surveyed historical examples of pop-up books and deployable forms to generate a library and design vocabulary of kinetic mechanisms [11, 94, 136, 233]. This work developed a mechanism pattern language, but does not allow for arbitrary recombination [233]. Thus, the bulk of user research on pop-ups has concentrated on supporting CAD modelling to facilitate experimentation with the mechanisms [11, 94, 166, 241]. Programming behavior consists largely in designing a given form, fabricating it, assembling it, and then activating the mechanism to test whether the behavior operates as intended [165, 222]. This line of research creates techniques for control over kinetic behavior, but ignores the use of kinetic behavior for tangible interaction and programming.

Paper User Interfaces

Paper user interfaces (PUIs) use tangible manipulation to leverage familiarity with paper artifacts to achieve proprioceptive control over computation [227]. PUIs locate interaction on paper itself, instead of a screen, so that manipulating paper's features affords manipulating structured data [109]. Pen-based interactions naturally supply affordances for manipulating complex data-structures familiar to paper [210]. By integrating sensing hardware into the pen form factor, unique marks and annotation create hyperlinks to dynamic media by serving as unique identifiers embedded within paper [12]. Similarly, printed identifiers—called *glyphs*—structure digital representations of paper as a UI hierarchy [109]. Yet PUIs encountered a midas touch problem where general paper interaction can be confused with intentional invocation of system behavior [226, 227]. The Digital Desk solved this problem by fusing microphone data with recorded images captured from an overhead camera to detect clicks [226], while PaperLink used a switch to trigger the pen's visual capture of paper-based information [12]. These techniques enable phrasing and chunking of sensor streams to facilitate novice manipulation of computer-identifiable patterns [40, 95]. These techniques enabled researchers to create a large design space for gestural input to PUIs.

Glyphs extend an object-oriented approach to PUIs. Originally, a lightweight approach to interface composition, glyphs collapsed a distinction between toolkit support for interface commands and that for application data [41]. As a result, the programming language basis for user interface widgets could be extended to user-supplied data, and so, user-defined commands [169, 12]. For PUIs, customizing the interface itself is of paramount importance. Paper documents persist despite tremendous advances in desktop computing largely because paper's accessibility supports defining and establishing idiosyncratic routines [142, 78]. Glyphs support social sharing of user created PUIS [93], syncing paper instantiations with their digital counterparts [88], adding audio interaction [175], and tangible control over digital data disclosure [132]. Similarly, 2D visual codes underlie a set of PUI widgets and make PUIs available through paper and mobile phone infrastructure [191], and successfully enabled secure financial management in areas with unreliable internet infrastructure for microfinance [172]. Glyphs enable PUI customization to facilitate hybrid digital-paper sharing of written and oral annotations with minimal computing infrastructure.

End User Programming Architectures for Paper

Some of these innovations in PUIs drew from or informed new architectures. The OOPS toolkit extended event-oriented architectures to handle ambiguity that arose from interaction techniques that needed to recognize and classify sensor data as an event [146, 145]. Up unto that time, most interactive system architectures supported a mouse and keyboard, but did not consider the ambiguity that arose when needing to support more natural input like speech or pen [146]. OOPs extended event dispatch with a set of objects called *mediators* with the purpose of handling ambiguity that arose through hierarchical event dispatch [146]. OOPS adopts a broad definition of a recognizer: a recognizer produces events that are interpretations of other data or other events [145]. Importantly, recognizers produce events that are dispatched through the typical event stream, and they store information about an event until mediators are able to resolve ambiguity that might arise [145]. In the OOPS toolkit, consumers return their interpretation of an event so that intermediate feedback could be provided to the user and other consumers or mediators could resolve the ambiguity in the case when previous consumers could not [146]. Mediators and their architectural integration allows for the developer to explicitly allow for ambiguity and draw upon a host of techniques to resolve the ambiguity either with more information or by consulting the user. For example, the raw data of an unclosed round stroke could be stored until a mediator is able to resolve whether the user intended to draw a circle or write the letter 'O' perhaps by asking the user to choose which was intended from a dialogue. Notably, OOPS was evaluated with several input devices and other interactions such as pen recognizers, speech recognizers, interactors-recognizer, and the Context toolkit [146].

2.6 Sustainable Socio-material Practices

Tangible interaction, and also, research on the Internet of Things could benefit from an ecological lens. An ecological lens involves analyzing the network of computing artifacts in the home to account for the people, practices, values, and technologies within an environment [73, 158, 110]. From an ecological perspective, the social and cultural context of an artifact's use can have consequences for assuming particular functions: that is, the likelihood that a vacuum cleaner will be used by a child in the household or whether phone notifications will be pushed to a laptop [73, 167]. The ecological perspective has parallels in tangible computing. Specifically, the ubiquitous computing—and thus IoT—agenda embeds computing in objects that have rich affordances to enable tangible manipulation of computing properties [105, 225]. The Tangible Interaction Framework built on these initial directions by situating tangible computing within a social user experience [100]. Individuation subsequently extended this framework and called attention to the routines, habits, and arrangements households use to shape a computationally enhanced object to represent or reflect relationships between household members to develop a research agenda on the social internet of things [9, 156]. Designs of EUP that have successfully worked within a community's dynamics recognize the role of computing in identity construction and relationships. To address these, they have adopted methods that make room for an artifact's role within community norms. These methods are often born out of the behavioral sciences and not cognitive science [157]. Two of the more influential methods for EUP are activity theory and distributed cognition [157, 142]. These methods have much more to say about how EUP can have a role in coordinating small groups. Thus, EUP for IoT positions IoT as an artifact based network that facilitates embodied manipulation of computing properties and construction of one's identity and relationships.

Engaged Computing and Seamful Design emerged to argue that ubiquitous computing's disappearance into everyday objects raises social and ethical issues [104, 190, 69], and importantly for this dissertation: sustainability issues. For example, the invisible actions ubiquitous computing takes on behalf of the user lead to deskilling and obscure important decision making features of day to day planning and action [69, 190]. In trying to be unobtrusive, seamlessness undermines the process of investing value in an object because there is no need to learn, adapt, and personalize it [9]. Further, invisible actions can have environmental costs by encouraging digital waste through use of energy and cloud infrastructure that a person doesn't actually need or want [177]. Routines are needed to sustain artifacts over time and help realize values of durability and sustainability [9]. IoT risks creating substantially new environmental and social costs if it ignores the routines and relationships computationally enhanced artifacts will be a part of.

For the past decade, researchers have begun to worry about computing's environmental costs. Through its preference for newness, the modern American household's excessive material consumption can have negative side-effects. These include increasing stress, deteriorating health, growing landfills, short life-cycles for non-renewable on non-biodegradable materials, and enlarged energy demands [14, 22, 152, 183, 209, 221]. A central theme of sustainable interaction design is the relationship between invention and disposal [22]. Researchers have tried to dig into this question to discover what objects Americans regard as their most valued possessions and what would they discard without second thought [152, 163]. Turning to second-hand consumption, they have tried to identify what role *newness* has in developing one's material identity and how does attachment develop over time [163, 174]. The results of these studies have led researchers to recommend design that can decouple ownership and identity, make reused objects fashionable, and facilitate DIY culture for object augmentation all with an eye to prolonging technology life-cycles and perceived durability [22, 163, 174]. By taking an upcycling approach, this dissertation shifts these questions to asking how can the invention of new technologies better support the tight-coupling between ownership and identity.

Significant progress has been made on systems focused on energy consumption, eco-feedback, and persuading home dwellers to be more environmentally friendly. Prior work developed a set of three personas–the helper, the optimiser, and the hedonist–to characterize the impact of identity on energy saving [108]. The desire for energy savings that can be central for upgrading to a smart home, can still conflict with other aspects of a person's identity and become entangled in household dynamics of compromising with other householder's desires [108]. Some household members lack agency and discretion to control smart home infrastructure. Socioeconomic class may in part determine whether residents are able to monitor their energy usage or fix a building's inefficiencies [63]. Further, since many household resources are shared, household members do not have the discretion to make unilateral decisions [48]. Rather, setting a device's state and controlling its usage is an interdependent decision [48]. Researchers have created display technologies to equip these uses with more information on what data has been collected, what the household's resource usage is like, and to further reasoning over what has happened in the home [42, 75]. One concern is that the way the household's resource usage is made visible to the user is underpinned (whether implicitly or explicitly) by specific models of behavior change and persuasion [75]. In contrast, end user programming (EUP) could enable users to investigate and reflect on their home's data or explore their own questions without imbuing the interface with particular persuasive techniques [42]. While revealing raw data streams might avoid assumptions of a one-size-fits-all model for the home, even an approach which reveals minimally curated data will still confront household politics and require research effort to determine what design requirements best accommodate household members' conflicting interests [42, 104, 47, 232].

2.7 Cultural Forms of Everyday objects

IoT devices are at risk of reinforcing undesirable social relationships within the home and across society. IoT will likely alter household routines, change control of living space, and even social conventions themselves, such as standards of cleanliness and good parenting [67]. Several studies found that screen-based technologies like video games, smart phones, and televisions undermine positive familial relationships when family members are unable to get the attention of others [2, 10]. Even when household routines are directly supported by new IoT devices, they may still reinforce traditional divisions of household labor that negatively impact specific classes and gender groups [232]. For example, middle and working class households differ in their uptake of new standards of good parenting that regulate children's use of devices [10, 56, 129, 159, 2, 234]. Similarly, family members frequently adopt gender-stereotypical roles with respect to who builds the home network and who learns to program the device [151, 189, 35]. The way IoT devices are incorporated in the home's division of labor can extend the reach of structural inequities like class or gender-based divisions in accessing and controlling devices.

The household uses embodied practices for managing and sustaining household life. Families use the home's spatial layout to manage activities such as private consumption of sensitive material, religious commitments, or quiet (although this varies by culture) [14, 49, 134, 221, 234]. In the United States, a large percentage of housing stock consists of older homes designed for labor and housework to be accomplished as a backstage activity [14, 2]. Nonetheless, kitchens function as command centers where families congregate to catch up with one another, coordinate, do homework, and collaborate over bills or school notes [49, 57, 51, 14, 213]. Parents use bathrooms to socialize children into good habits like hygiene and cleanliness [14, 2]. Bedrooms are private spaces that may be free from electronics or the internet altogether, or allow for consumption of specific content [49, 14]. Technologies can impinge on these divisions by violating house rules (*e.g.*, giving kids internet access in their bedrooms), or in cases like the TV, directly organize spatial layout [14, 49, 134]. By treating internet connected devices homogeneously, IoT disrupts the home's implicit organization and management.

Incorporating the home's possessions in an IoT infrastructure must be done with care. Objects and their life-cycles carry layers of social meaning [72, 81, 110, 1, 2, 221]. In other words, they are polyvalent [195]. Some objects may be discarded or destroyed not because of the object, but to sever the relationships they are a part of [1]. In the United States, household objects realize family ideals such as nurturing growth, talent, creativity, self-expression, and identity [2]. Support for these ideals is currently missing from IoT [91]. By using domestic objects as design material, a system can support reinventing the home's existing socio-material contexts to align with its values.

Interactive Books

Books are a promising, lightweight interface for the home. Their cultural form invokes desirable social practices that can support collaborative exploration and shared decision-making [98, 99]. When Vannevar Bush described how computing could provide lasting societal benefit, he envisioned a new kind of book facilitating creative linking of the physical world with the wealth of knowledge contained in a library [39]. This idea animated *Dynabook*: dynamic media, the size of a notebook, responsive to the user's wishes by facilitating creative expression and association through the *Smalltalk* programming language [113]. Despite this initial interest in books, decades passed before the form factor emerged. *ListenReader* explored the physicality of reading by embedding electric field sensors in the spine of a book to detect its handling and an RFID reader in the cover to detect the currently opened page with RFID tags [17]. The form factor facilitated collaborative interaction through social reading [17], multi-generational accessibility through familiar paper affordances [55, 176], spatial mapping of audio for indexing through writing conventions of the book's target language [206], and maintaining a record that synthesizes photographs, data, tracings, and even scraps collected from the physical environment through pasting, co-locating, and layering [143]. Tape, glue, and stickers enable users to physically link external information and objects to books [143, 93]. This has been supported in interactive books through printed barcodes [143, 118], stickers with fiducial markers [93, 99], hand-writing the related code [143], and pen-based interaction [143]. Like paper, interactive books leverage familiar affordances, but also integrate with social norms by evoking social practices of collaboratively creating and annotating a shared model of the physical world.

Interactive books offer promising alternative abstractions for trigger-action programming. Paper pop-up mechanisms housed in a book form factor can support users with manipulating and triggering sensors using tangible abstractions [179]. Similarly, sticker-based interactive books can support tinkering with circuits [180], peel and stick construction of circuits [97], and hand crafting remote messaging [74]. Sticker composition can introduce programming concepts like sequencing actions [99]. These books borrow techniques from GUIs that use shape to constrain permissible compositions in the language [99, 182]. For example, slots in a book constrain how physical cards can be inserted [212]. Paper tags, such as visual codes, enable dynamically mapping triggers to actions to co-evolve alongside familiar scrapbooking practices in a cognitively accessible form [132]. These paper mechanisms can thus support proprioceptive control over the material environment in a way that designs for agency in new technologies [132]. Paper mechanisms promise to facilitate a wider audience's exploration and expression of computational ideas by concretizing an EUP language's abstractions. By leveraging familiar paper engineering techniques, interactive books provide a suite of paper mechanisms that can expand participation in end user programming by attending to the material constraints that make some actions available while excluding others.

Emerging research on interactive books introduces a suite of techniques for rethinking what books could do for sketching in hardware and supporting creativity. Researchers embedded bend sensors, micro-switches, and speakers into a book-like interface to investigate navigation techniques afforded by books such as page flipping and page reserving as ways to skim, bookmark, and retrieve visual content [223]. *Qook* uses a projected display and AR fiducial markers to provide keyword searching, highlighting, and bookmarking without instrumenting the book itself [238]. Books' dependence on writing conventions' spatial mapping supports sequential storytelling techniques by mapping manipulation of paper-based fiducial markers to parts of a visual story [66, 99]. Augmented reality books employ pop-up visualization techniques to enable dynamic content to emerge from the page [20], support direct coloring of digital content through templates [50], and facilitate learning complex, physical models [149]. Complimentary, books expanded tangible interaction techniques: paper-mounted electronic modules produce paper located outputs [33], pop-up mechanisms support manipulating and triggering sensors [179], sticker-based hardware supports peel and stick construction of circuits [180, 97], and wireless sensors and actuators in the form of *I/O Stickers* support functioning, remote messaging interfaces [74]. Sticker composition can introduce programming concepts like sequencing actions by borrowing visual language techniques that use shape to constrain permissible compositions in the language [99, 182, 181]. Similarly, slots in a book constrain how physical cards can be inserted [212]. By leveraging familiar paper engineering techniques, interactive books provide a suite of paper mechanisms attentive to the material constraints that make some actions available while excluding others. This dissertation builds on this prior work by incorporating PUI and interactive book techniques in The IoT Codex to show how kinetic mechanisms open up a new space for end user programming for domestic IoT.

3 The Upcycled Home: Removing Barriers to Lightweight Modification of Everyday Objects

3.1 Introduction

Internet-of-Things (IoT) devices promise to enhance even the most mundane of objects with computational properties by seamlessly coupling the virtual world to the physical. Yet, IoT research to date has largely focused on designing wholly new devices, while ignoring many of the existing objects in current households. Instead, we propose an upcycling approach. Upcycling is the process of reusing an object by transforming it into something of greater value or quality. An upcycled IoT would enable users to upgrade the home by transforming their possessions into IoT devices. This approach complements how users already acquire and relate to their objects. Even before the advent of personal computers, ethno-archaeological studies found that over one third of household objects enter the home in used condition [194]. More recent work on product lifecycles uncovered the importance of object attachment, appropriation, reuse, and lateral cycling for how people treat their possessions in the long term [163, 174, 221]. Recent innovations open the opportunity of this approach with battery-free, wireless sensing [138, 205, 89]. We build on this work by using this opportunity to make use of the home's possessions as design material for IoT systems.

An upcycled approach to IoT confronts several barriers. First, family members have unequal availability to participate in IoT decision-making. They have different schedules, different skills, and different stakes in the process. Second, not all *families* have the same access to smart home systems. Structural factors, like renting a home, limit some families' power and autonomy when integrating these systems into their household [63, 147]. Third, IoT systems are not always compatible with the way families' manage and use their possessions. For example, families exercise room-level control of their objects (a bedroom TV is used differently than a living room TV), but most IoT systems homogenize how objects are treated across different home spaces [67, 68]. Finally, upcycling objects with IoT requires families to imagine new, technologically-augmented uses for their belongings. This type of creative reimagination is possible, but not always easy to achieve [30].

These barriers align with existing challenges for IoT systems. For one, existing systems struggle to incorporate meaningful collaboration, especially when family members have different levels of contribution [35]. Most research to date studied relatively affluent families or other early adopters. Additionally as noted above, systems homogenize by residence rather than by room, undermining families' mental models of how home works. Finally, IoT requires novel ideation techniques when working with users to envision their future homes [59].

This chapter seeks to address these barriers to an upcycled IoT. As a formative study, it employs home tours and semi-structured interviews to uncover how households organize objects in their daily lives and the domestic roles sustained by them. The findings contribute 1) 3 models for how families coordinate household labor and work, 2) a user study focused on the needs of families who experience forms of structural marginalization, 3) a characterization of room-level object management practices, and 4) a characterization of how families project their desired home onto their possessions. The results demonstrate opportunities for IoT to support lightweight modification of existing object forms and, through those forms, existing social relationships.

3.2 Methods

This study uses ethno-archaeological methods and samples participants according to intersectional identities to uncover the routines around domestic objects in diverse households as given by race and ethnicity, gender, age, disability, and class.

Ethno-archaeological methods underpin a material culture approach to studying the modern household. Material culture studies assumes the presence or lack of artifacts reflects human behavior [183], and how space is used can speak to loci of coordination [14]. These methods developed during the late 20th century as archaeologists turned to landfills and garbage cans to infer household consumption patterns [14]. Famous among these, the Garbage Project gave birth to a burgeoning field known as garbology and examined the relationship between Americans and the objects they discarded [14]. The researchers found that contrary to American stereotypes of consumerism and waste-making, only 6.2% of costly items were discarded (*e.g.*, refrigerators, ranges, and washing machines) while most were reused [183, 194]. These trends changed during the personal computer era as electronic devices began to enter the household. Yet, they show that domestic objects may tell an alternative story of household activities than previously supposed. We adopt this line of inquiry in our study to examine which objects could be upcycled and which objects should remain in their current state.

As for this study's approach to sampling, recent scholarship has called for representing users using an intersectional approach to address society's systemic and structural forces that may result in research contributions differentially impacting subpopulations, particularly those deemed vulnerable [197]. In a meta-review, the authors argued that gender and gender fluid categories intersect with race and class in important ways that deserve greater attention when recruiting and reporting on participant demographics [197]. Further, they argue that analyzing research findings using a single critical dimension such as gender, or race, or class, can obscure important features that reveal how society's structural inequalities can have a compounding effect because the analysis only considers a single structural factor [197]. Taking this critique seriously, this study purposively samples participants for a representative user group according to six categories of structural marginalization: gender, race, class, disability status, age, and household structure. This study uses these specific categories because a representative distribution of the general population could be validated against publicly available data, and these six categories are known to exhibit concerning inequalities with respect to research findings improving opportunities for a high quality of life.

3.3 Participants

This study recruited 10 households for our study from a mid-sized American city (Pittsburgh, population 300,000). It sampled participants according to 6 criteria: gender, age, race, class, disability, and household structure, in line with persistent categories of concern [65, 197]. It excluded potential participants when, (1) working with them would require changing the study's protocol (*e.g.*, require a translator), (2) their household had changed over the past year (*e.g.* new baby), (3) their household had <2 people, or (4) the study had already recruited enough similar households. This shifts the study's population away from a representative sample. In the sample, 100% lived in the same home a year ago and averaged 2.9 persons per household compared with the city average of 77.8% and 2.1 persons [38]. The study's analysis characterizes children <5 yrs. and those with severe intellectual disabilities as part of their households, but they were not interviewed or directly worked with. NGOs and public organizations helped with recruitment using word of mouth and sharing the study's flyers. Participants that were eligible were required to commit to 1) the entire duration of the study, and 2) having all of their household members participate. Many family types such as divorced families, or gender fluid households, were not successfully recruited even though they were advertised to through related organizations. Their omission is a limitation.

3.4 Procedure

Carnegie Mellon University's institutional review board (IRB) reviewed and approved this study's procedures before it commenced. The study took place in participant homes, and sessions generally lasted between 1.5-2 hrs. One household lasted 3 hours due to disability related delays and interruptions from unexpected visitors. Participants were consented according to IRB protocols and children 5-18 years old assented with parental approval. Compensation for adult participants was \$15/hr and children >5 years, \$5/hr as a token remuneration for their time and effort.

The study lasted 7 days. On the first day, participants completed a demographics questionnaire, gave a home tour, and completed a one-on-one semi-structured interview. Home tours and interviews were conducted in parallel. Those household members who were not being interviewed gave a home tour to a member of our research team. Thus, children gave home tours twice. In one family with 3 adults, the family gave 2 complete home tours and 3 one-on-one interviews.

Daily Activity Interviews

Building on prior work [2], the protocol employed a semi-structured interview asking adult participants how members divide the households' main activities and their awareness of others' activities. During the interview, members were separated since participants are more likely to honestly disclose about their partners in their absence [24]. Participants were asked to describe a typical day, which activities take the most and least time, others members' activities, whether they participated, and activities they wished their household spent more time on.

Home Tours

The protocol elicited decisions on upcycling domestic possessions by adapting home tour methods capturing participant attitudes towards their home [2]. Participants were asked to show 5 rooms in their home. In each room, participants were asked to choose 3 objects to modify with computing abilities, and 3 objects that they would not want to modify. Next, participants were asked to explain their reasoning behind their choices.

3.5 Data and Analysis

Audio data from the home tours and interviews were professionally transcribed, and survey answers were digitized. Grounded theory methods were used to analyze home tour data. On the first pass through the data inductive codes were developed and deductive codes applied when the data supported previous findings such as attachment [1, 163], parenting and technology [10, 159, 234], reuse [174, 183, 194], and ownership [14, 2, 1, 152]. On the second pass, two research team members discussed and clustered the 107 codes into 3 categories on domestic artifacts' role in division of labor, network management, and household acceptance. Analysis proceeded to axial coding with the emergent 3 themes and 11 subthemes, collapsed overlapping codes, and developed new codes when fit was imperfect. Finally, one research team member uninvolved in coding thus far, used the 3 themes and their 11 subcodes to code two randomly chosen transcripts. They were then debriefed for coverage and characterization of the data.

This study uses portraiture to portray and contextualize findings on the relationships sustained by domestic objects in diverse households across race and ethnicity, gender, age, disability, and class differences. Portraiture sketches the connections between participants' individual personalities and organizational culture by portraying their authority, wisdom, and perspectives [131, 130]. This method centers their views within careful ethnographic description so they might be fully recognized, appreciated, respected, and scrutinized [131, 130]. This descriptive process resists grounded theory's abstract portrayal of theoretical concepts supported by grounded description and their tendency to subsume the lives of some participants under the voices of others. It does so by infusing findings with the color and contours of participants' embodied lives as they touch on the theoretical concepts and by balancing theoretical clarity with the uncertainty of attending to features that threaten conceptual simplicity. In this work, portraiture is used to counteracts unequal representation by being forthright about which participants' socio-material practices are being described and which family's data is employed to reach conclusions.

One-on-one interviews were coded according to field notes and the interview protocol. 10 daily activities were co-constructed with each of the 20 interviewed participants and the protocol's 3 themes (one's day, awareness of others' day, and desired activities). Activities were clustered and collapsed according to their overlapping codes. This resulted in 15 activities. Family members' responses were then compared to identify activity patterns across households. The results are presented using portraiture and pseudonyms for families and their members.

Indicators of Class

Income Ranges[a]	*American City*	*Study Pop.*	*Households*
<$10,000	12.4%	0%	0
$10-14,999	7.5%	0%	0
$15-24,999	12.5%	10%	1
$25-49,999	23.9%	10%	1
$50-74,999	16.5%	50%	5
$75-99,999	9.8%	10%	1
$100-149,999	9.8%	10%	1
$150-200,000	3.6%	0%	0
>$200,000	4.3%	10%	1
Education Level[b]	*American City*	*Study Pop.*	*Participants*
<High School	8.1%	0%	0
High School	27.6%	5%	1
Associates	7.9%	15%	3
College	21.3%	30%	6
Graduate	19.4%	35%	7

[a] For our 8 married households, the comparison for married couples may be more appropriate: 26.6% <$50k; 19.2% $50-<75k; 15.4% $75-<100k; 19.9% $100-<150k; 8.5% $150-200k; 10.3% >$200k

[b] Based on adults >25 years old; For the household of adults <25, comparisons with adults 18-24 years would be appropriate: 22.3% bachelor's or higher (in our study, 100% of this 1 household).

Table 4: The table shows the representation of different classes in the local population relative to the households represented in the study's population. We used income and education to characterize the class structure of our population in line with standard conventions using income, education, and job type.

3.6 Findings

Working with 10 households, I found that household members bring society-level constraints home and work with other family members to renegotiate their approach to ongoing demands made from both inside and outside the home. I identify 3 division of labor patterns that illustrate how households realize society's structural inequities, and thickly describe how these are embodied in home life. Against these background patterns, I found that family members use domestic objects and spatial layout to set and enforce the home's norms. They create boundaries in relationships and instruct others in domestic roles and responsibilities. Lastly, household members contend with existing, nuanced networks of ownership when making critical decisions to incorporate IoT into the home. We characterize these ways that households regularly reuse and appropriate domestic objects to create, sustain, and reconfigure their relationships. These findings inform what support is needed for IoT decision-making, and illustrate ways that upcycling could support piecemeal upgrades to the home.

Below, I first summarize participant demographics. Then, I present the 3 division of labor patterns, and follow up each with a family portrait. Next, I present 6 family portraits that show how families use objects to develop shared mental models of home and how this can be in tension with current IoT. Objects are specialized to their containing room and are used in compliance with those rooms' norms. After each portrait, I draw lessons from each household to inform management and control over an IoT system and design support for family members integrating IoT in a way that makes progress towards their aspirational home.

3.6.1 Demographics

I recruited 29 household members resulting in 26 participants (42.9% male and 57.1% female). I had slightly more female representation than the recruitment city (51% [38]). Participants ranged between 9 to 70 years of age (M=35.8 yrs., SD=20.2 yrs.; City MD=32.9 yrs., [36]). The recruitment city had the following age distribution: 5% $<$5 yrs., 15.8% $<$18 yrs., 70.2% adults $<$65 yrs., and 14% adults $>$65 yrs [38]. Our study's age distribution approximated this with 8.7% $<$5 yrs., 34.8% $<$18 yrs., 65.2% adults $<$65 yrs., and 21.7% adults $>$65 yrs. Seven household members reported having a disability. This prevalence is at times higher than the city's: 9.1% $<$65 yrs. (compared to 9.9% for the city) and 83.3% $>$65 yrs. (compared to 13.6% for the city) [38, 36]. We summarize further in Tables 4, 5, and in prose (sources [36, 37, 38, 164]).

3.6.2 Division of Home Labor

When comparing households, I observed highly integrated morning and evening routines. All families were together for dinner, and almost all, during the morning routine. This does not mean that all families ate meals together. In many families with children, children ate on an earlier shift than parents. Morning routines tended to be asynchronous with points of contact between family members due to differing rising times, bathroom scheduling, or calculated prep times for children, pets, or

				Race & Ethnicity					
	Citizen	*English 2nd Language*	*Hispanic*	*Amer. Ind./ Alaskan*	*Asian*	*Black/ Afr. Amer.*	*Hawaiian/ Pac. Isl.*	*White*	*Two/More Races*
City Pop.	91.5%	10.8%	2.8%	0.2%	5.5%	24.3%	0.0%	66.3%	3.2%
Study Pop.	75.9%	24.1%	13.8%	3.4%	10.3%	27.6%	0.0%	37.9%	6.9%
Households	8	2	1	1	1	3	0	5	2
				Household Structure					
	Female Headed	*Male Headed*	*Married Opp. Sex*	*Same Sex Couples*[1]	*Children*	*Neither Par. Employed*[2]	*Mother not Employed*	*Father not Employed*	*Both Par. Employed*
City Pop.	29%	8%	63%	1.4%	44.1%	3.8%	26.3%	7%	62.9%
Study Pop.	20%	0%	70%	10%	40%	25%	25%	0%	50%
Households	2	0	7	1	4	1	1	0	2

[1]Same sex couples city comparison figures are for the entire US. [2]Employment city comparison figures are for the entire US.

Table 5: The representation of different racial and ethnic groups, household structures, and working parents in the local population relative to the study's population. The number of households present in our study for that subpopulation is given below the percentage.

others with a disability. I identified 3 patterns—*Cruise Control*, *Labor Specialization*, and *Balanced Awareness*—that are characterized in more detail below.

Cruise Control

Cruise Control families listed under half of their routines in common. This style is called *Cruise Control* because, compared to other families, participants rarely mentioned household management. They do chores, but did not seem to manage the process. These families rarely, if ever, mentioned any hobbies or exercise. They worked through lunchtime and multitasked: doing homework or answering e-mail. Their life styles exhibited asymmetry. One family member described a single, additional activity omitted by their partner, while the other listed several (>3). In one family, many differences arose from the head of household living with a significant disability. In the others, one family member was stretched thin balancing many side jobs, while their partner worked long hours. These partners were employed professionals in a field requiring a graduate degree and had guaranteed, predictable and steady hours. For 2 families, one partner described the other as doing chores, while the other described the first as playing video games. These couples desired more time to relax together.

The Walker Family Portrait: Celine and Mia are a young and energetic, married couple who own their 2 story house in a suburban neighborhood on the edge of the city. They make twice the median income of their surrounding neighborhood (average for the city). It is over 85% white and has >75% home ownership. Their home has brightly colored walls lined with meditative sayings or photos of the couple together and is populated by several dogs. Mia describes her and Celine's routine a year ago when Mia had a single 9 to 5 job. They had a date night when they would go to a show together or go for a walk on the waterfront. Now, Celine is busy with 3-4 jobs and caring for her relative with a cognitive disability.

Often, Celine and Mia's schedules do not align, so they cherish their weekends and dinners together (after Mia comes home and before Celine goes back to work). Celine spends the most time preparing for these:

> *We're always trying some different diet that—I'll be in the kitchen for four hours a day. The worst one was when we did raw veganism, and I was literally in the kitchen food prepping for five or six hours every single day, because everything has to be fresh.* (Celine)

Upon arrival home, Mia is drained: *"Emotionally... I bring it home.... There's a lot of really horrible things that happen to people."* Alone in the evenings, Mia watches TV while researching home renovation. The Walkers consulted a contractor about installing a dishwasher, but halted their plans when the level of structural change meant renovating the kitchen.

Pattern Lessons. Cruise Control family members often work on the home or prepare for collaborative activities in isolation from one another. Their asynchrony limits familiarity with each others' activities. Job demands constrain their availability and energy to invest in collaborative decisions. An upcycled IoT should support these families by enabling hand-off of prep work and minimizing project creep into deeper structural changes to decrease coordinated decision making.

Labor Specialization

Labor Specialization families listed half their routines in common. They described little exercise and few to no hobbies. Unlike Cruise Control families, they have a high division of labor. One member functioned as the 'manager'. They track the home's state and direct attention to critical needs. Breakdowns occur if this person forgot since others did not always recognize when they should contribute. Although most family members mentioned chores, the manager described the check-in process when chores would be divided. Family members knew each others' habitual chores (partner verified), yet, felt they were never-ending. Their activities frequently diverged and included multiple activities omitted by others (>2). Labor Specialization parents wanted their children to do more chores, eliminate cleaning up after them, and hasten house work. Families without children had tight schedules accommodating a particular life stage's needs like school or rehabilitation. These families desired a shared effort at cleaning and organizing their home.

The Martinez Family Portrait: The Martinez family lives in a rented townhouse in a wealthy suburb occupied by >90% white families, >$75,000 median household income, and >75% home ownership. Their neighborhood is clean and friendly. Located across from a golf course's lushly manicured lawns, it is near a park offering several recreational options. Julio is college educated and commutes to his IT job in the city. His wife, Carmen, describes herself as a stay-at-home mom, but confesses to sometimes working remotely for her privately owned business in her country of origin. Carmen regularly prepares the family's breakfast and dinner, walks the pets, does laundry, picks up her kids from after-school activities, grocery shops, and helps with homework. Carmen describes how there are breakdowns when she loses track of the household.

> *With the day to day, trying to work and cook and take care of them, sometimes I forget. It just seems that they never take—if I don't walk the dogs, feed the cats and the dogs, it seems something very common. If I'm not here, if I'm not on top of it, nobody feeds them.*

Carmen describes Julio as proactive in helping around the house, but she is responsible for knowing what needs to be done. Julio earns the majority of the household income and often comes home after everyone has eaten dinner. His family will sit and catch up with him while he eats. When dinner is finished, Carmen and Julio load the dishes and put food away together. Then they join their kids watching a Netflix[3] show.

Pattern Lessons. Labor Specialization families take a divide and conquer approach to house work. Doing so enables the family to parallelize tasks and complement each others' contributions. To leverage this collaborative process, an upcycled IoT should enable setup and maintenance tasks to be subdivided into parallel processes and make each members' role transparent and easy for others to learn. Thus family members could rotate a managerial role or swap roles so that task specialization does not become an entrenched routine.

Balanced Awareness

Balanced Awareness families substantially overlap their activities (7 or more). Family members check-in daily at a prearranged time. They delegate errands to a specific time to correct for likely forgotten items. Chores are swapped. Or, the one person who regularly does a chore—cooking or networking the home—specifically enjoys it. These families have significant hobbies, like gardening or singing, that ground their members' self-conception. Yet, they do not try to do too much and described only 1-2 unique hobbies. This allows time for household upkeep, work in parallel, or to be available when needed. These families desired spontaneous or unstructured time to get outdoors and break with their routines.

The Baker Family Portrait: Janel and Joshua Baker own their 2 story house in a racially diverse suburb of the city. Their street is lined with tall trees and yards with children playing. Making well over the median income for their surrounding neighborhood (a little below the city average), they are from the city and so, have family nearby. Janel and Joshua were high school sweethearts and had their first child when they were just out of school. They recently had a second child—a daughter—ten years later. Five days a week, Janel and Joshua work full time

[3] Netflix is an online television streaming service.

outside of the home and are home together in the evenings. They divide dropping off and picking up their children while commuting to work.

At the end of each day, the Bakers check in to see whether dinner is on track. Did they remember to defrost what they had planned? Should something be picked up from the store? When they arrive home, Janel multitasks in the kitchen. She does this during the interview—feeding the baby as she speaks—and explains that transitioning to dinner takes time.

> *I come home after sittin' in traffic, and once I get home we can talk about our days—"Hey. How's your day? How've you been?" Talk to the baby. Then I get ready to make dinner... once I make dinner, then we feed her.*

Janel sometimes socializes with friends or attends board meetings for a local association rather than return home directly after work. Likewise, Joshua goes to the gym and will periodically bring their son Caleb with him. Describing his weekends, Joshua smiles, *"So cutting grass isn't supposed to be relaxing; but sometimes it is, because you're just outside."* He finds ways to enjoy even chores.

Pattern Lessons. Balanced Awareness families have integrated routines or management strategies that are resilient to breakdowns and surprises. Making time to coordinate and accomplish housework is not a problem. Yet, these processes are so established, they undermine spontaneity and experimentation. To support these families, an upcycled IoT should nurture creative ideation and role play. Then household members could try new household arrangements to stretch the family to grow.

Summary

3 patterns emerged for coordinating housework within the confines of family roles and commitments. Cruise Control families invest in the home asynchronously and need an upcycled IoT to support project hand-off and limit scope creep. Labor Specialization families have a manager who directs members to work in parallel and complementary roles, and need support with swapping roles and assuming responsibility. Balanced Awareness families switch roles and scheduling as needed, but need support in escaping routine. These typologies characterize how families currently allocate time and attention to jointly accomplish housework. To complement these, we developed pattern lessons an upcycled IoT should use to support families with trying out new arrangements.

3.6.3 Negotiating Social Boundaries through Ownership

Family members use objects to instruct other members in household norms. During their home tour, participants emphasized their relationships to their objects or their object-mediated relationships with others. Most household objects are functionally shared between all family members. Yet, ownership and authority are regularly used to cue, negotiate, or control relationships between household members. Objects are used by households with children to construct and enforce rules of behavior as part

Division of Housework		
Cruise Control	*Specialized Labor*	*Balanced Awareness*
High level of care giving; Sometimes disability present	Significant commute to work; stay-at-home moms	Large overlap of routine activities; High diversity of activities
Little mention of chores or errands; no management	Ownership of chores; Manager of the house	Chore and Errand swapping
Stretched thin; Lunch non-existent	Housework manager directing attention to tasks	Dividing tasks in half with integrated roles
Almost no mention of exercise & no mention of hobbies	Diminished exercise & little to no hobbies	High level of exercise and 1-2 significant hobbies that shape identity
Asymmetry in awareness of others tasks; Video games or watching tv and other does chores	Aware of the division of labor and verified recognition of specialization on certain tasks	Often are available or watchful for ways to jump in to complete tasks
Desire quality time together or time for a special occasion	Desire collaborative effort at chores so that the group has the ability to do something fun together	Make time for special occasions; Desire unstructured time, spontaneity, and getting outdoors
Jameson, Walker, and Taylor Families	Martinez, Carroll, Gilmore, and Chaterjee-Basu-Mistry Families	Baker, Olson, and Crane Families

Table 6: The above table show the themes that shape and distinguish each of 3 household routine patterns found in the 10 households we interviewed. 3 household patterns were identified by comparing adult members responses with other members of their home.

of nurturing child development. Even in households without children, objects are used to set boundaries, signal consideration, and coordinate tasks. Acquiring and discarding objects presents a cost, as displaced or discarded objects disrupt these time-earned negotiations.

Owners have Imaginative Authority

Knowing how to behave towards other family members' objects and rooms is part of knowing the rules of the home. Many objects are shared, but a select few belong to a single person. Eighteen participants identified objects specifically belonging to themselves or others in the home, and 12 participants emphasized when objects were shared.

The Jameson Family Portrait: Janice and Tameeka live in a rented townhouse in a neighborhood occupied by over 90% African American residents with a median income of <$25,000. Janice is Tameeka's grandmother and is unable to work due to a disability. Janice is relatively young for her grandmother status and glows when talking about Tameeka's projects and involvement in a neighborhood program for at-risk youth. Tameeka, age 12, proudly shows off 'her room' with 'her TV'. When adding computing abilities to her room she explains, that she would start with the *"simpler things"* such as her floppy-eared, stuffed rabbit, or her giant bear. Tameeka would add IoT services that could enable her stuffed animals to talk. For her, IoT could help her bedroom's imaginative world come alive. Tameeka and Janice have a relationship that Tameeka describes as *"awesome"*. However, she is careful to consider when she has crossed the threshold into her grandmother's domain. When giving a tour of the house, Tameeka giggles at the opportunity to violate household norms by making unsupervised use of her grandmother's room: *"Finally I choose her room!"* Tameeka does not want to upcycle her grandma's closet. She explains, *"her closet is perfect for me to play hide and seek in if she would let me."* Her grandma's closet nurtures Tameeka's imagination, but her freedom to enact her fantasies in that space is limited by ownership.

Object Lessons: Personal objects, like Tameeka's stuffed animals, realize and sustain their owners' imaginative ideas. The personal process of adding computing to these objects enables owners to project their fantasies onto their world. Upcycling should enable owners to encode their imaginative ideas into upcycled objects during setup.

Room Lessons: Rooms have owners. Through ownership, family members have authority to make the room's rules and use its boundaries to instruct others in its norms. An upcycled IoT could respect this practice by enabling the home setup to be subdivided and customized at the room level.

Claiming and Enforcing Territory

Owners personalize and claim territory to signal their wishes. Conflicts over objects arose during 7 home tours. However, ownership conferred authority to enforce a person's preferences to resolve conflict.

The Carroll Family Portrait: Nicholas and Sara Carroll are both college educated. They married after going to school together, and then relocated to the city because of a job opportunity for Nicholas. Sara quit her job as a school teacher to stay home and raise their three children: Josh, Caleb, and Tyler. On the home tour, Josh and Caleb decide that it is important to keep the lettering of their names on their bedroom wall untouched by IoT. When asked about their reasons, Caleb declares "Territory!" Josh, who shares the room, echoes the sentiment and provides more explanation:

> *It claims territory when our friends come over... Also if my brother's about to touch [my things], I can say stuff like, "Do you see that name above there; that's there for a reason!" and he'll back away.*

These territorial claims are not unique to children. Sara claims territory too. She unhappily explains that her lamps should remain unaltered by IoT, but they are currently broken. They were a casualty of Josh and Caleb playing football in the house. Sara uses the football and lamps to reinforce her point to Josh and Caleb as she gives the tour: the football belongs outside the house, and the lamps were not Josh's and Caleb's to break. Josh and Caleb violate the home's rules by misusing her belongings. Sara recounts how Josh and Caleb buried her wedding silverware in the backyard dirt as treasure for their game of pirates. She still feels the loss, as no one has been able to locate the buried silver in the yard of their rented house.

Object Lessons: Fixed objects, like Caleb and Josh's wall lettering, create stable rules for a room. In contrast, roaming objects (*e.g.*, football, wedding silver) move throughout the home and so the room rules governing them vary. An upcycled IoT could work with a spectrum of object types by supporting fuzzy object properties that range from stationary, fixed behavior to roaming, in flux behavior.

Room Lessons: A room's owner uses its objects to signal the room's rules. Shared spaces without clear owners (*e.g.*, living rooms or dining rooms) are sites of conflicting values since the room's rules are negotiated among the household's members. An upcycled IoT should defer room level policies to the negotiated arrangements and provide for dynamic change over time. To do so, policies could be set by the objects at the focus of attention and prioritized according to social hierarchy.

Negotiating Boundaries

Objects successfully or unsuccessfully enforce social boundaries by expressing the owner's identity or limiting sharing in a relationship. 9 participants used objects to support their self-image, their household role, and interaction with others.

The Gilmore Family Portrait: Tyler and Chloe Gilmore are a retired couple who own their three-story house in a low income neighborhood of the city. Retired now, Tyler worked as a computer programmer in the military. He protects his home by wanting to upcycle his current alarm system by adding more security features to the window using IoT. The Gilmores *"do a lot of cruising"*, and he worries an intruder *"might come in and break the window while [they're] gone...[or they] might forget to lock the windows"* Tyler portrays himself as a protector who safeguards the family and home. He worries when Chloe and their daughter are home alone. Chloe recently recovered from surgery and is limited in her ability to get around. The Gilmores installed a motorized chair on their stairwell to enable Chloe to move between floors on her own. Showing it off as she descends the stairs, she exclaims, *"Thank God for the stairlift."* It inspires many of her IoT modification ideas. She would add IoT to her possessions so that she could use them independently or remain in her own home as she ages. Chloe would add IoT to her bath so that she can bathe alone:

> *For those of us who have disabilities...It would just make it more friendly and [sic] wouldn't have to call on other family members or somebody to keep you company. You'll [sic] more independence...it also does something great for you when you are able to do it by yourself.*

Chloe's husband frequently assists her bathing, but she feels a sense of accomplishment and dignity when doing it herself.

Object Lessons: Objects—like Chloe's stairlift—mediate family relationships by modifying a room's norms. When objects cannot be used independently, they breach these norms and create asymmetrical relationships as household members require others' help. An upcycled IoT could address these failures of objects to sustain independence by recommending augmentations of those objects so that they can be used independently.

Room Lessons: Rooms are perceived as hospitable or inhospitable. They can signal the home's boundaries to outsiders by using surveillance to cue transgression. Rooms alienate insiders when they breach household norms. This becomes more salient as the space becomes more personal (*e.g.*, a bathroom). An upcycled home can incorporate these norms by taking advantage of the privacy gradations implicit in the home's spatial layout. Greater control over the IoT system could be made available in insider spaces where only a privileged few have access and access could be limited in spaces available to outsiders.

Summary

The home's spaces are both private and shared. Their norms structure family interactions and discriminate insiders from outsiders. Owners have the authority to set norms for how their part of the home or possessions are used. These possessions mediate the relationships and ground their dynamics. They can be fixed to a room or roam between room-level jurisdictions. An upcycled IoT should defer to negotiated norms for the home's spaces and possessions. It could do so by aligning system access with spatial privacy, recommending object modification patterns, supporting a spectrum of fixed to roaming object types, and enabling room-level partitions and policies.

3.6.4 Modifying Objects to Create the Aspirational Home

Domestic possessions carry prior expectations from the way they already work in the home. Some of these possessions are regarded as essential to peoples' lives and so, are non-negotiable. New IoT capabilities compete with these prior arrangements and must engage with them. For many households, the home and its construction is a given. These assumptions constrain what the home can accommodate or adapt to. Yet, new computing modifications enable new arrangements that evolve the household closer to its members' ideal home.

Making a Household Work

The home's objects are sorted according to those that function reliably and those needing continual upkeep and repair. Family ideals for the home praise functioning items, because they *"[do] what [they're] supposed to"* (Miguel Martinez). Seventeen participants thought computing-enhanced objects could disrupt or restore their ideal home by introducing greater fragility to routines or automating upkeep like to-do lists.

The Olson Family Portrait: The Olsons are a technology-savvy, retired couple who outfitted their 2 story house in a residential neighborhood with cloud storage and remote access to their music collection. During her career, Sheila managed a database system and is considered *"the techie"* (Gordon's description). Sheila creates many custom bill-pay, medication management, and party planning systems for their home. Though retired, Sheila feels she works most days.

> *Processing stuff, trying to keep up with stuff, trying to understand things that come in... we got a new car at the end of May and I asked [the company] to send me the booklet. And I never got it, so I call her and she said "Oh, I'll send it again", and I never got it, so that folder sits on the table, waits, so I have to call her again. (Sheila)*

Sheila's systems process household information and automate housework. She teaches Gordon how to use them, and he knows exactly how to input his part so that he can hand the bill payment off to Sheila. He is wary of modifying household objects with IoT knowing the amount of time they already spend on maintenance. He explains that they're *"very fearful of Windows 10"* because they are forced to upgrade Sheila's custom systems. He worries about investing in these costs.

Object Lessons: Workplace object practices extend work to the home by sustaining management routines. Even when experts successfully teach them to other family members—like a bill pay system—these practices nurture unwelcome psychological strain like exhaustion. Domestic IoT should disguise computing techniques borrowed from the workplace to help household members distance themselves from their jobs.

Room Lessons: Even when IoT supports information access across room boundaries, spatial arrangement—like a folder on a table—is still used to cue family members on the state of the information system and next steps. Thus, an upcycled IoT can use place and spatial layout to provide system affordances that are idiosyncratic to each household's social dynamics.

Which Objects Should Be 'Smart'?

Fourteen participants want computationally modified objects to give them peace of mind, realize their commitments, and provide reassurance. Objects that are successfully interwoven in the home supply this support, and are valued as a result.

The Taylor Family Portrait: Dave and Katie Taylor are a young, married couple who met in college and relocated to the city so that Dave could pursue a graduate degree. They live in a recently built, one bedroom apartment located in a rental complex as part of a suburban, shopping district with a population that is 80% white. Dave and Katie personalize their home with an extensive collection of games they regularly play together and decorate it with posters from their favorite books and movies. Before Dave gets home from school, Katie begins cooking dinner. When he arrives, he chops the vegetables or washes the dishes while the food simmers as she directs him to. Many times Katie worries. Has she left the stove on? What about the toaster oven? She wishes she could remotely cut off power to parts of her home. She describes a recent storm occurring while the couple was away from home: *"We lost power a couple nights ago. We were worried—did we fry our TV? Did we fry our game system?"* She worries about damage to the entertainment hub they've invested in and that support the family's leisure time together. Similarly, Dave worries about damage. He wants to ensure they keep the apartment to the company's standards. He wouldn't add computing abilities to the floor, walls, and doors. They should remain exactly as acquired so that they do not have to pay the rental company.

Object Lessons: Play and leisure support personalization, customization, and connection with others. In doing so, they enable families to create ownership of rental and temporary spaces. By centering play and leisure, upcycling could enable families to customize a temporary space into a smart home.

Room Lessons: Families do not have full control over the rules governing temporary or rented spaces. These rules constrain the structural depth to which computing can be integrated. Yet, families desire room-level management of computing capabilites, and need those same affordances available through their possessions. To support this, upcycled objects could set room boundaries and so, function as a room's walls.

Mental Models of Home

Family members develop nuanced models of their objects' roles within their household's flow. These models limit members' ability to explain decisions to add IoT to some objects over others. For 8 participants, the decision was obvious: part of how they conceive of the object categorically.

The Crane Family Portrait: Lisa and Kevin are a married couple on the brink of retirement. Both have graduate degrees that they used in their professions. Lisa already retired, but Kevin still works full time at a nearby hospital. They own their 3 story house in a wealthy, residential neighborhood (median income >\$120,000). Kevin enjoys music, and his interest is clearly expressed throughout the house. When entering the Cranes' house, visitors walk by his 3.5 feet drums in the foyer. The

Cranes learned how to audiocast music to their decades old, classic speaker system with the help of their son. They arrange and modify their household to nurture their interests. Lisa loves to cook and spent the past three years planning and remodeling her kitchen. She wouldn't want to alter her cookbooks with computing abilities because she dislikes online recipes and prefers the physical cookbook.

> *Because it's a book. You can handle it. You can mark it. You can see it. You've got history there. It tells a story. There are stains from the recipes you've used a lot. You mark it—this works, or that—change that. I can leave something for my kids. (Lisa)*

For Lisa, upcycling the cookbook with IoT implies a digital screen that she couldn't spill things on, or use to record the recipe's history. The Cranes treasure the object forms that they have selected and shaped over the years.

Object Lessons: Families nurture their interests by investing in domain-specific possessions like drums or a cookbook. These objects ground creativity and talent. Upcycling's added value should sustain these investments and entrenched uses, yet encourage domain growth like audiocasting did for the speakers.

Room Lessons: Owners use their control over the home's rooms to structure support for inventive activities such as playing music or cooking. In this way, the home itself buttresses identity building and formation.

Adaptable Objects and Essential Objects

Participants' life stage informed which objects should be upcycled and how (9 participants). Some objects shouldn't be learned anew during early and late life stages, demanded too much time, would quickly be outgrown, or required too much responsibility. These objects needed to compromise with big life changes like babies, graduate school, or a new disability.

The Chaterjee-Basu-Mistry Family Portrait: Roommates Yasmeen, Meethu, and Neha live in a two bedroom apartment in university housing. All three are college educated and currently pursuing graduate studies. Neha has lived in the apartment for longer than Yasmeen and Meethu and occupies the solo bedroom. She describes the kitchen's continuous disarray: *"It gets really messy, and there's a lot of space crunch. Since we're students, we leave the house early in the morning."* The family has limited time and space at home. They neglect kitchen objects to accommodate demands made from outside the home, or reconfigure their livingroom to accommodate guests such as a family member or boyfriend. They desire more control over their apartment:

> *This window is really small. It blocks all the sunlight, usually. It feels like I'm trapped at times...It just pisses me off. That's the reason why I can't get an air conditioner, because I do not have the window space.* (Yasmeen)

Yasmeen learns to accept the window as is and instead, would upcycle its decor. Modifying the blinds would *"be a fun thing to do."* and liven up the window she resents. The family feels empowered when they successfully work within the apartment's constraints. Beaming at their ingenuity, Meethu shows off the shelf the family assembled to hold their foodstuff.

Object Lessons: Adaptable possessions, like reconfigurable furniture, give owners control over their home's constraints. Since many families, like roommates, are together for a short time, their shared objects do not carry timeworn negotiations. Upcycled objects should use adaptation to harmonize competing desires for household norms (*e.g.*, kitchen upkeep) by supporting changeable functions throughout their lifecycle and making them intelligible so that they may be renegotiated.

Room Lessons: Rented rooms can result from compromising housing with growth needs for specific life stages. As non-ideal, they emotionally impact inhabitants and undermine aspirations for home (*e.g.,* Yasmeen's window). An upcycled IoT could nurture positive associations by helping owners' reinvision their aspirations for home through customization.

Summary

Families invest in objects and rooms to nurture their creativity and growth in varying degrees based on control over their space. Family members want to structure their space using IoT to better nurture their growth. Yet, because of their limited control, they need the ability to use object-level infrastructure to function as room-like to accomplish these goals. It should adapt to rented spaces to honor renters' commitments to owners, have affordances capable of harmonizing multiple owners' wishes through reconfiguration, and sustain inventive processes rather than migrate workplace management into the home to ensure the home is restorative and relaxing.

3.7 Discussion and Limitations

This work found that families use objects to adapt the home's space by setting and enforcing norms. Modifying domestic possessions allows families to project their ideals onto the household and adapt rooms to nurture creativity and growth. To support this, families need IoT infrastructure to range from room-centric to object-centric change. It should accommodate norm setting that dynamically changes across spatial jurisdictions and temporary owners (*e.g.*, renters or borrowers). At times, participants were wary of IoT's disruptive costs like displacing routines, discarding functioning items, or making skills obsolete. To minimize these, an upcycled IoT could support lightweight modifications of the home's relationships by preserving object forms and using them to ground infrastructure.

Preserving Form and Managing Displacement

At the outset, this work argued that family members do not have equivalent availability to integrate IoT and that IoT impinges on mental models of home. Its findings showed how an upcycled IoT could leverage families' object-practices instead. Doing so could enable households to tailor IoT and make it more accessible to mental models of home. By contributing to family members' self-conceptions and their relationships to others, existing possessions are accessible to preconceptions for how they, as objects, should work when modified with IoT. As a result, differing family members could make IoT decisions according to their control and understanding of an upcycled artifact. Households evolve idiosyncratic arrangements over time and construct family roles through object-norms. Introducing new IoT devices risks displacing these negotiated relationships. In our study, objects were strong boundary markers—especially in families with children—for household customs. Lamps regulated children's behavior indoors, and a cookbook's material properties crafted a family legacy. New IoT interactions could respect these customs by preserving both existing object forms and relationships (*e.g.*, with others, object attachments, etc.).

Yet, family members are not always happy with the home's current arrangements. Objects can reify problematic relationships and remind members of painful history. For example, objects obligate family members to others or require help to effectively use. IoT costs are not simply monetary. Instead, costs incur from the disruption IoT brings. With the new interactions made possible by IoT come displaced processes, requirements to upgrade or reconstruct past practices, and the work of configuring the new technology to the household.

Upcycling domestic possessions could aid in redefining the home. Modifying objects with IoT could support constructing new relationships and crafting ideals. For example, upcycled objects could re-allocate family members' time and attention. Domestic possessions would not obligate the family 'manager' if they could convey their own priority, proper use, or messages from other family members. An object could even reassure a person that it is not, in fact, a priority. Earlier studies found that married heterosexual women have a heightened awareness of household chores and could benefit from diminishing expectations for household organization and cleanliness [24, 2, 159]. Upcycled home objects could change household conventions, like cleanliness standards, by shifting responsibility, providing a check on perceived needs, and avoiding increased standards associated with new technology. For example, Sheila Olson's folder could be upcycled with messaging capabilities by piggybacking on a standard protocol like e-mail through the wireless communication afforded by an RFID lightbulb as described in [89]. Cleanliness is one convention of many embedded in day to day life, but it illustrates how upcycling could shift entrenched household norms.

Customization Instead of Discarding

Earlier, we claimed that an upcycled IoT should support family members with reimagining their possessions with computing capabilities. We found that participants' concern with discarding objects in working condition creates an opportunity for an upcycled IoT to address household values of minimizing waste by envisioning a new life for those possessions. Many participants were reluctant to acquire new objects and thought IoT would require discarding those they currently enjoyed or were "perfectly good". In many cases, these object worked just fine and participants thought it wasteful to discard it. Domestic possessions were investments families weren't willing to ignore or write off. By including these objects, an upcycled IoT alleviates some of these worries and could support making improvements on objects that merely "function well enough". For example, the Cranes were delighted their classic speaker system could be part of new, audiocasting cabilities. In contrast, the forced upgrade to Windows 10 undermined the previous investment the Olsons had made in building their custom, organizational systems. Often boredom with an aesthetic genre, or a want of agency, novelty, and self-expression results in a desire to rearrange domestic environments [1]. Enabling customization of an upcycled IoT would support making deliberate decisions about which properties to discard while keeping those that are satisfactory. These could be added at any point during an object's life cycle to refresh older objects with new capabilities just as rearranging furniture or adding a new coat of paint renews and recreates a room [1].

An upcycled home could support personalizing objects that are inherited, rented, or passed on from their previous owners. In out study, many domestic possessions saw second and third owners through family inheritance or changing roommates. Households used personalization techniques to adapt objects to their new owners or current point in time such as claiming a side of the room, manipulating a recipe's tastes, or expanding food storage space. An upcycled IoT could use these techniques to help new owners adapt previously owned objects to their own tastes or needs. Just as recipe modifications and wall-mounted lettering were used by our participants to personalize their possessions, upcycling could facilitate digital naming and annotation. Or, it could enable object versioning to allow users to modify an upcycled object's parameters but retain previous owners' choices. These techniques leverage the IKEA effect by enabling owners to use modular configuration or assembly of pre-designed adaptations to household possessions [162, 35]. Users can then adapt pre-designed IoT modifications to their objects to reinvision local constraints.

Limitations

This study investigated 10 households in one American city, but this is not enough to fully characterize needs for lightweight modification. Other cultures should be examined. They will undoubtedly use objects and space differently [221, 35]. It remains an open question whether the division of labor typologies and family portraits presented here would adequately speak to these alternatives.

3.8 Conclusion

This study worked with 10 diverse households to shape an upcycled IoT to minimizes risks of destabilizing domestic relationships and values, and to characterize the home's object focused practices. It portrayed 3 patterns of how households divide labor to meet competing demands made from both inside and outside the home. These patterns show how societal level constraints are embodied in home life and prefigure potential costs of IoT. Across households, findings showed that domestic objects are used to negotiate social boundaries, nurture growth and adaptation to constraints, and make progress on an aspirational home. These results identify several household niches where IoT could support lightweight modification of existing object forms and social relationships through upcycling. An upcycled home would support customization to give users control over which object properties will be modified and how disruptive the modification will be. Further, it would give family members the ability to manage the costs of newness such as what will be displaced, discarded, or made obsolete. This work contributes portraits of household niches amenable to upcycling.

4 Understanding Family Collaboration around Lightweight Modification of the Home

4.1 Introduction

Upcycling can extend computing to everyday objects by adapting computing capabilities to existing possessions. Domestic possessions need not be replaced with internet-enabled equivalents. Instead, upcycling can support their renewal by retrofitting them with the latest computing capabilities. Within this vision, upgrading to a smart home consists of augmenting domestic possessions with an internet connection and related IoT services. Domestic possessions could then persist with an adaptable user interface capable of accommodating their existing forms. By being grounded in the home's already existing networks of relationships, values, and routines, these upcycled objects could offer families greater discretion and control over upgrade costs.

The previous chapter contributed portraits of household niches amenable to upcycling everday objects with IoT. These portraits characterize how households embody societal level constraints through the home's division of labor, ways in which everyday objects are used to set norms, and the strategies households use to nurture identity construction but also compromise ideal life with current demands. These thick descriptions prefigure the introduction to IoT in the home and sketch opportunities for IoT to be adopted in piecemeal fashion by leveraging the home's existing socio-material practices. However, these portraits did not chart out the adoption process. While prior studies have delved into the details of how households integrate new technologies within everyday life, there is little understanding of how families might upcycle their everyday objects.

The work reported in this chapter sought to understand how families might make lightweight modification of everyday objects in the home with computing capabilities. Whereas chapter 3 reports findings from portions of the first day of a 7 day study working with 10 families, this chapter reports findings uncovered from the entire duration of the study. Using participatory design, households were asked to enact the process of modifying their home objects over 7 days by attaching stickers to their possessions and endowing those objects with computing capabilities. This study found there are three principal facets to the household's process to modifying domestic possessions: generating ideas, contesting and reconciling a shared model of home, and planning and programming the home. From the findings, this chapter also summarizes trends in household modification of everyday objects and presents results on household ratings of the costs for a prospective *IoT Sticker* system from a User Burden survey.

IoT Stickers

Upcycling proposes an alternative vision for IoT: use the household's everyday possessions to enable household members to make 'dumb' objects 'smart' by attaching a wireless, battery free sticker to the object. Recent developments in RFID interaction techniques, back-scatter signal detection, and at-home fabrication suggests this vision is possible through a class of technologies I call *IoT Stickers* [15, 77, 89, 112, 127, 138, 205, 230]. These technologies are approachable to the typical household through RFID-based sticker technologies with widget-like interaction techniques such as clicking a button or sliding a scroll bar [138, 205] and installable with lightbulb form factors for RFID readers [89].

This work extends this prior research by systematically investigating how households might tailor IoT stickers to the domestic environment using an experience prototyping method with sticker props [202]. Following prior work [103, 144], this study uses the IoT Sticker button-like interaction technique to explain to participants how a tailorable IoT system might work and to enable them to enact their design ideas in a domestic setting.

4.2 Methods

Over 7 days, an in home study was conducted to uncover household approaches to using IoT Stickers to retrofit domestic possessions with IoT services. To do this, households were asked to consider which objects to augment with IoT capabilities and enact this process using endowed sticker props. The approach is described in further detail below.

Procedure

The study took place in participants' homes and lasted 7 days. This consisted in a 1.5-2 hour home visit the first day, a diary study the next 5 days, and a final 1-1.5 hour home visit the last day. Scheduling needs were accommodated by relaxing the study timeline to end between the 7th and 10th days. One household's child did not participate the last day despite the study's requirements. Participants were consented according to Institutional Review Board (IRB) protocols, and this included assent from children 5-18 years old with parental approval. Each adult participant was compensated \$15/hr and each child $<$5 years, \$5/hr.

Home Tour and Sticker Task

On the first day, different mixes of family members conducted two tours of the home. For families with children, this meant that each parent led one tour while the children participated in both tours. For one family made up of three roommates, this meant two roommates at a time led each tour, with a third roommate swapping out for different portions of the tour. This allowed us for interviewing each adult in isolation, remain within the allotted study time, and collect shared perspectives of the remaining family members concerning household objects. During the tours, family members showed us five rooms. In each room, they selected three objects that they would want to modify with computing abilities and three objects that they would want to keep in the same condition without augmentation. They were then asked to describe their augmentation and reason for selecting the object. This encouraged family members to envision the upcycle process, and supported collecting data on objects inappropriate for IoT.

At the end of the first visit, participants were instructed on how to enact a sticker modification and complete the audio diary portion of the study following an experience prototyping method. First, participants were asked to select an object, describe the new abilities they would give to that object, and to provide a detailed characterization of how the new capability would work or change the environment. An audio-recording USB stick was left with each family, reminder instructions, and a pack of Avery office stickers to use as props. A practice entry was facilitated with participants before the study session ended. Participants were reminded to make a diary entry each day through a text message for each of the subsequent five days. During the last session, participants were asked to demonstrate their process of making an entry. Next, participants were asked to show their diary objects and to describe their modification process in case of device malfunctions and to capture shared household assumptions undescribed in the audio diaries.

Semi-structured Interview

The family was interviewed about how the sticker task went using a semi-structured protocol [133]. Questions included asking how the process went, how participants generated ideas for their objects, and how much time it took.

User Burden Scale

Following the interview, each participant was asked to fill out a user burden questionnaire. The questionnaire adapted the difficulty of use, privacy, and financial dimensions from the User Burden Scale to suit questions for an IoT sticker system [208]. Privacy questions were framed in terms of current policies on commercial IoT devices. To measure user sensitivity to device cost, *IoT Stickers* at $0.10 and *IoT Lightbulbs* (required as a reader for IoT Stickers) at $300 from previous research [89].

Analysis

All in home sessions were video-recorded an the audio data from the diaries and home studies were transcribed by a third party transcription service. Because of device malfunction, the audio diary data is unavailable for P5. For the home tour data, qualitative codes were developed for objects participants modified, those they kept unmodified, and their reasons for doing so. With the results, a dataset was created of objects that included who tagged the object, what room it was in, its modification code if any, and a reason code. The findings present descriptive statistics for general trends in this dataset. The other transcripts were then analyzed using thematic analysis to characterize families' process of tailoring an IoT Sticker system to their home. For the user burden questionnaire, findings are summarized using descriptive statistics.

P1's data is excluded from most of the analysis because the last session and the audio diary data revealed that the household did not use the sticker props to enact the upcycling process. P1 is included in the object modification dataset and the user burden survey results. This exclusion is discussed further in the discussion.

4.3 Findings

Overall, families grappled with three principal facets—idea generation, the model of home, and planning/programming home—of tailoring an IoT Sticker to an object. To generate ideas, participants needed to distance themselves from their environment to consider how they might modify their possessions. The section Idea Generation characterizes this distancing, how objects seeded ideas, and the environment's function as a source of ideas. This section also describes trends in participants' initial ideas to tailor IoT to their possessions. Next, the study found that object modifications potentially destabilize the family's shared model of home, and they can serve to stretch the family to consider new arrangements never before considered. The second section summarizes this. Finally, families do substantial work—including coordinating and collaborating on proposals—to get buy in, implement their ideas, and foster trust in how the home will change. The section Planning and Programming Home characterizes this process. Below, participant demographics are described before the findings are presented.

4.3.1 Idea Generation

Modifying an object with IoT begins with coming up with an idea. At times, the object itself is the source of inspiration. However, households also used other sources such as caring for a family member or wishing to completely change a room. When generating an idea, families decide which objects are appropriate interfaces for IoT.

Choosing Objects for Upcycling

The dataset revealed 17 different rooms and 267 objects as candidates for IoT interfaces. Participants designated a total of 219 objects to leave unmodified with IoT. The largest number of objects to modify were in the bedroom (53), bathroom (52), living room (42), kitchen (35), and office (18). These 5 rooms contained 74.9% of all modified objects and 77.2% of all unmodified objects (see Figure 2 for a breakdown of objects for each room). Nine rooms were chosen by only one or two households: guest room, baby's room, dining room, door/doorway, hallway, outdoors/porch, sewing room, TV room, and storage. These rooms fell into three categories: liminal spaces, such as hallways and doorways; dedicated activity spaces, such as rooms for sewing or watching TV; and spaces for other people, such as a baby or guest.

Bedroom. The bedroom contained the most selected objects in the study (modified: 53; unmodified: 52). Many cherished objects, such as gifts, resided in the bedroom. Participants thought these obvious targets to keep unmodified by IoT. Participants were attached to many objects that connoted comfort and relaxation. For example, the bed was the top item to keep unmodified in the whole study (9 participants). P1b explained, *"It's very comfortable...I have seen those*

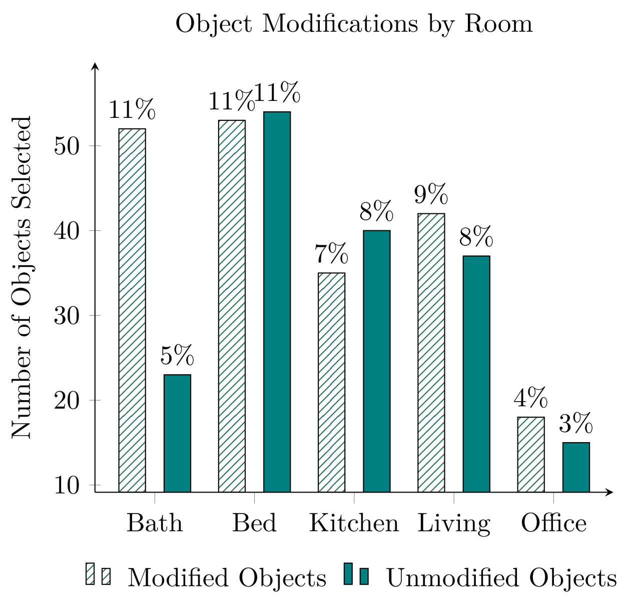


Figure 2: The bar graph shows the distribution of objects across the five rooms with the most number of selected objects from participants. Each room shows the comparative number of objects participants wished to modify with IoT capabilities to those they wished to remain as they are, apart from an IoT network. Percentages above each bar show the percentage of the total object inventory from the study found in each room.

beds that rise up with a remote...but I think my bed is great how it is." Participants described their furniture as aesthetically appealing and fitting well (4 unmodified). The bedroom TV had the most divided opinions (5 modified, 4 not). Participants modified the TV with upgrades to fit fashion: networking, mounting on the wall, or to simply have one in their room. Others thought their current TVs worked well, didn't think modification worth the cost, or weren't appropriate for a child's room (P5b). They modified lights to be turned off from the bed (6 participants) and the closet to manage items and recommend fashion choices (4 participants). The bedroom contained the most number of cherished items that were inappropriate IoT interfaces, but also contained many objects which could change the environment to be more comfortable and helpful.

Bathroom. The bathroom was the only room with an unbalanced number of modifications: over twice the number of modified items to unmodified (52 v. 23 unmodified objects). In contrast to the bedroom, participants were little attached to bathroom objects and there were few they wouldn't modify. P4b dismissed bathroom objects, *"No, there's nothing we want to keep—no, we'd change the whole thing, change it all out."* Most often, objects were to be added (9 participants) or replaced (4). They targeted the shower, toilets, and tub for renovation. The top reasons to modify an object were to increase comfort of the room (6 participants) and to improve mood (8). Two of the top 3 objects to remain unmodified were part of plumbing: the sink already worked well (5) and some had an aversion to any kind of modification to the toilet (2). In general, participants wished to not simply augment existing objects, but completely renovate their bathroom.

Kitchen and Living Room. The kitchen and living room's objects led to divided opinions across family members on how to modify shared objects. Family members disagreed over modifying the refrigerator (manage the food inventory or attach screens; 7 participants) or whether it already functioned fine (unmodified by 6 participants). In 4 of the 8 households, family members in the same home disagreed. Similarly, some family members found the stove frustrating to use (3 modified) and others thought it worked well (5). Notably, two children did not want the stove to be modified because they were just learning to use it and that was difficult enough (P5c and P5d). In the living room, the television inspired indecision (5 undecided). They wanted to modify it to fit with fashion (8; see bedroom TV discussion), but were also satisfied with how it functioned (7). Unlike the bedroom, living room furniture was not always kept unmodified (5). Changes included adding missing items (3) and making them more accessible or easier to use (2). However, many participants were attached to their existing furniture (9) and features of the room itself–floating shelves, chimney brick, or the room layout–because they thought they were aesthetically pleasing. Like the bed, the couch was considered an object of comfort that shouldn't be modified (4 participants). In the living room, almost half of the objects selected as inappropriate for IoT were so because they were aesthetically appealing (16 of 37). Family members disagreed the most over modifying shared objects in both the living room and kitchen.

Finding Ideas

Families generated ideas for modifying an object using methods such as brainstorming, problem solving, and defamiliarization. Problem solving has been covered elsewhere [151, 35], so the findings focus only on brainstorming and defamiliarization here.

Brainstorming. Family members took turns describing their ideas to the rest of the family. P9 rotated ideas by day, *"We each had ideas. Oh, that's a good one, so we'll do mine today, and we'll do that one for tomorrow" (P9a).* During this time, family members listened to each others' ideas, generated extensions, tried to envision them, and clarify what others had in mind. P10's ideas evolved from one anothers',

> *P10b:* We would collect...on some days and decide on an object—and we'd spend half an hour brainstorming as to which object we were going to select for that particular day, and how's it going to work, and once we had a rough structure ready, then—*P10a:* We started answering the questions, and more often than not we came up with totally different systems—*P10c:* With new ideas, and then we started recording.

Brainstorming was a time to ask questions and empathize with other's envisioned changes. Although participants were instructed to record their entire process, brainstorming usually happened off the record. P3a wanted her family to be perceived as competent; "This was our process—talking about it before we sound stupid." The study team learned about brainstorming during the last day's session when participants were asked to walk through their sticker modification process.

Defamiliarization. To generate ideas, family members began questioning their household. Defamiliarization enabled members to reflect upon their familiar environment and reconsider domestic possessions anew. To do this, *"[P9a] just walked through every room looking at everything."* Participants would cast about the room looking for objects to modify, walk from room to room searching for material inspiration, and at times, reconsider their ingrained routines. Three participants went through each room in turn. The room strategy helped with systematically exhausting room based sources of inspiration: *"We did the mailbox, the pantry, the bathroom. We haven't done anything in the basement. Is there anything we could do in the basement?" (P2b)* Occasionally, walking itself would stimulate reflection on a behavior or object. P4a *"walked out of the room. Came back. [P4a] said, 'Oh. I left the light on.' And by the time [he was] going in and out of the room and coming downstairs"*, he had generated an idea for modifying the light. As several days passed, participants began to view their objects differently while going through typical activities. One day during the study, P9a was reading a book on the porch. She then realized that she frequently dozes off, and she decided to add a sticker to her lounge chair to set an alarm to wake her up after an hour. Participants were able to generate ideas through distancing themselves from their environment.

Defamiliarization pushed family members to reflect deeply on their lives. P4a explained that, *"It sort of let [him] reflect to see what was actually going on in [his] home"*. Two family owned homes struggled with modifying their household objects because of the years of invested effort tailoring it to their needs. P9 *"had the house setup pretty well as it is. It's pretty organized, functional in its layout...[They] were just thoughtful over the years."* The time of owning and living in their own home benefited the household's arrangement. In contrast, defamiliarization diminished P2's—an immigrant family living in a rented townhouse—view of home. *"Then you start to see your house as very archaic, you know? You're like, oh, this is bad. I don't like that" (P2b).* Defamiliarization helped families reflect upon their values for home, but could also surface problems that were not always something a person wished to focus on.

Summary

Families generated ideas for tailoring IoT interfaces to their home by using objects and rooms as sources of inspiration during brainstorming or defamiliarization. Intimate spaces like the bedroom and bathroom were targeted as sites for modification with IoT, but felt that objects connoting comfort and aesthetics should remain unmodified. Objects in shared environments, like the kitchen or the livingroom, generated disagreement over whether the object should be modified at all and what form modification should take. Yet, families found ways to work through modifications. Brainstorming was a collaborative process for empathizing with others and evolving new ideas, while defamiliarization was a solitary process that pushed families to reflect on how their values were realized in their home. The process of idea generation called for suspending belief in the current home to consider reconfiguration.

4.3.2 Contesting and Reconciling the Model of Home

During idea generation, families articulated a model of home that shaped how members interacted. In doing so, participants delimited the roles expected of one another, society's imposed constraints, and their views on what might be both technically and socially possible. Modification ideas could serve to destabilize the model of a functioning home. Specifically, object modifications mediated these boundaries by constraining the discussion or offering an alternative domain through metaphor.

The Family's Idea of Home

To explain their ideas, family members articulated a well functioning verses a dysfunctioning home. They designed object modifications to get rid of dysfunction. If P6b modified the oven, *"then the house won't smell like fried chicken"*, or P4b's modification *"would eliminate [the blinds] being broken or crooked and having to continuously replace them."* Modified objects would make the home function better. When P5b excitedly told her family, *"there's a button next to [the toilet] that you can just press and it'll sanitize it for the next person"*. With this change, P5b described her better functioning home as one where she didn't clean the toilet. Object proposals supported family members in changing the shared idea of the home and its arrangements.

Equipped with an object idea, a family member would then build consensus or buy in from others for the proposed modification. Even if the household generally went along with one person's ideas, family members would check in with others to see whether they would use the newly modified object. Would they benefit? P10a asked, *"What if we had this in the fridge? If we had this feature then what would we use it for?"* Check-in also tested whether others would use the object in the envisioned way, or appropriate it. P5b explained, *"He and I discussed completely different ideas for each thing, too. We had completely different—I like this and this, and he had different reasons."* By reaching consensus, family members created the boundaries and implicit constraints of the household.

Some objects challenged the shared model of home and shaped change. Usually, these modifications directly targeted household roles and not the objects themselves. For example, both P2b and P6b—stay at home moms—wanted to modify objects in the house that they didn't want to be responsible for any longer. P6b confessed that living with 4 males (2 young boys, and 1 boy potty training) meant she cleaned the toilet daily, and she found it disgusting. She modified the toilet to clean itself. P2b frankly told her family that they should modify the fridge: *"To waste less, and to think less, because I find it very, very tiring to think about what I am supposed to do for you."* These modifications revealed to other family members that the household member thought it problematic that they were expected to care for some objects in the home.

External Models and the Home

Societal norms constrained object modifications. These norms set expectations for the home and members' roles. In turn, these norms informed the features of modified objects. Participants used norms to justify properties like pay per use:

> *There would also be built in [the oven a] command system for paying when you're ordering like that, so that they can get your money. They're not just going to send it on our word.* (P6b)

Other norms included parents cooking healthy meals for children (P2b), or apartment residents being responsible for purchasing their own equipment to guard against power surges (P7b). Family members used norm accommodating features to free up their own time and attention. For example, P2b's imagined refrigerator provided guidance to her children on how to create their own lunches with the food available. P2b was happy to be relieved of teaching her children how to do this. Guidance could come from other sources as long as the children had enough help to proceed independently. Using norm modifications, family members offloaded societal expectations to the household's objects. This meant family members would no longer be personally required to intervene, but they could be assured the norms were upheld.

External models aided family members with proposing a household reconfiguration. Participants pulled ideas from hotels, restaurants, grocery stores, and public restrooms to import these external functions into the home. These places functioned as reconciling metaphors that illustrated proposed changes to others. For example, P2a worked to clarify P2b's idea asking, *"It's like Walmart, right? In the house."* Public places illustrated how similar kinds of objects existed elsewhere, but organized social relationships differently. These external models filled gaps in explaining how the object modification would work and what changes it would entail.

Determining What is Possible

Family members diverged in how much they constrained their object modifications by what they thought was technically feasible. Three family members became worried about whether the ideas could be implemented, and notably, they also had a career in a technical field (a systems administrator (P5a), a database administrator (P8a), or a systems analyst (P2a)). They worried some ideas stretched the limits of possibility, and could present obstacles to object ideas. P2b thought his family members *"All sort of know what the problems are and what we would like to have, but we didn't know if it was possible"* (p2b). Coming to a shared sense of what would be a feasible modification was difficult for many household members, and caused them to remark that they might be just engaging in *"wishful thinking"* (P8a), or struggling with *"first world problems"* (P9a).

Other households and family members were content to imagine something new and grew attached to their ideas. These families gained confidence by nurturing an alternative vision of home: *"We realized we're both technical geniuses"* (P3a). These families enjoyed approaching their house as a design material. P5b began to consider other ways to limit screen time in children's rooms, and remarked, *"I had never thought of shutting the Wifi off in certain rooms only."* The process pushed families to address household issues differently. One household directly disregarded technical feasibility. P6a disagreed with his wife, *"Oh I've seen enough Star Trek. If they can do this, and they can do that. Why can't [we]—you know?"* When families wrestled with determining what they could possibly implement in their homes, they stretched the family's shared model to include alternative household configurations.

Summary

Each family elaborated a model of how the home worked by implicating their relationships when modifying objects. Object modifications served to critique the family model by highlighting ways it could be better or is currently dysfunctional. By changing attributes of the objects that were central to the family model, family members created alternative roles and relationships for themselves. Societal expectations and norms shaped the family's model of home. This external model constrained the home model, but also suggested desirable alternatives. When considering possible modifications, families destabilized the current working model and extended it to include previously unimagined alternatives.

4.3.3 Planning and Programming the Home

While a few families only had one member create object modifications, most families worked on the modifications collaboratively. Working together required time, planning, and coordination throughout the work week. The majority of time spent on modifying an object consisted of coming up with an idea. Collaboration could fuel this process, but required family members coordinate their schedules to do so. Even if family members did not directly collaborate, part of maintaining trust included clearing ideas with family members and ensuring their views were represented in the modification.

Taking Time to Modify the Home

Participants spent most of their time on coming up with ideas for household objects. Although we instructed participants to document their ideation process in the audio diaries, none of the households fully did this. Participants reported that coming up with ideas took anywhere from one minute to two days. Four households admitted that for the first few days, they had no problem coming up with ideas, but then they hit a creative wall. This pushed participants into taking time outside of the recorded time of creating an audio diary entry. They used this time to reflect on their own, bounce ideas off other family members throughout the day, or delegate responsibility to others. Excluding this ideation time, participants reported the sticker task took between 3-45 minutes. Our audio diary data revealed that participants took a maximum time of 10.5 minutes to create an entry, and suggests that the recording was usually prepared in advance.

Planning Modifications

Except for two households, families planned and coordinated their modifications collaboratively. Four households found collaborating on an idea fun by introducing new dynamics into their relationships. P3b described how the activity changed her interaction with her partner: *"I would get home a little bit earlier so we could do the audio diaries before she went to sleep. We would have conversations that we never had before."* Three households described coming to a creative halt where they struggle to think of new ideas.

> P9a: *The first three days were kind of fun...Then that fourth day we hit...*P9b: *It seemed like it became more of a chore, and it was hard to think creatively about what else we were going to do. It became more like homework.*

These three families used a process of modifying objects that they had always wanted to change or that emerged during the first day's study interview. Once they had documented their legacy ideas, they struggled to generate new ideas. Collaborative approaches seemed to guard against this halt. One family changed their groupings, *"We always sort of switched...I think its a different dynamic, also."* (P2b). Interleaving different pairs helped larger families generate alternative approaches and new ideas.

Coordination was important for building the rest of the household's trust in the system. Three households used spare moments throughout the day to ensure everyone's views were represented, even if that partner would not be home to participate in making the audio diary. Coordination led to greater certainty for family members not present.

> Even when I wasn't home and not doing the diary. I knew what [P7b] was going to be doing and already kind of understood what he would be talking about. So it wasn't like [he] made this new function for our house and didn't tell [me] about it. Like, it was there. I knew what was going to happen. (P7a)

Households that used coordination over collaboration for planning, largely delegated the task to one household member rather than do the task collectively.

Programming the Home

Once decided on an idea, the family needed to shape that it into a workable modified object. To do this, participants desired the ability to manipulate features of the object's new properties to adapt it to its context. Newer families (together <10yrs.; 4 total) directly wanted to program their modifications with a voice interface or smartphone. They described manipulating parameters such as setting a timer or creating if-then relationships that let them control household objects remotely through buttons on their phone. Older families (5 total) focused on complex processes that were already part of their ingrained routines such as making grocery lists, seeking out recipes, preparing dinner, cleaning the toilet, upgrading software, and adjusting the lighting at different times of day. These families tailored objects according to subtasks that were already part of their typical activities, but the modifications allowed them to offload the work to the object or distribute it across multiple people instead of one. Workable objects for newer families drew from newer user interface paradigms, but for older families drew from entrenched activities or routines.

Local Leads

Patterns of family dynamics could be observed in the audio diary data. In every household, one member assumed a central role—call this the *local lead*—in creating object diary entries. There were 3 characteristic patterns that this member assumed: specialist, manager, and leader. For households with a specialist member (2 households), they created almost all of the diary entries for that household. Theirs was the only voice in the data, and they provided the most detailed description of modifications. The data suggests that these specialists worked in isolation to generate an idea and work through its implications. Manager households included other family members in the audio diaries, but the manager was the only person who was consistently present for most of the sticker entries (4 households). They facilitated a complete description of an object idea such as making sure the object was given a name and fully specifying how the idea would work and/or modify the household environment. Lastly, for households with a leader, object modifications were guided and directed by these people (2 households). Much like a specialist, object ideas tended to be that of the leaders and so they also provided the most detail and worked through the idea's implications. Yet, unlike specialists, leaders did not work in isolation, and usually all household members were present. Unlike household managers, leaders did not facilitate other members' ideas or work to

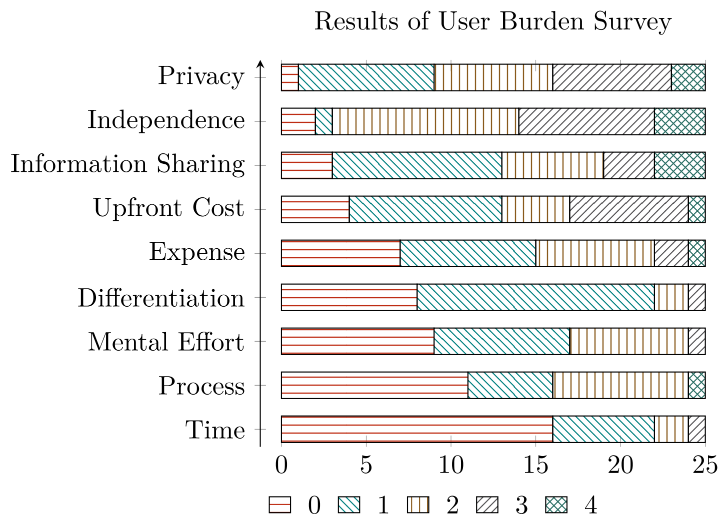


Figure 3: The stacked bar chart shows the distribution of family members' responses to the adapted User Burden Survey. The survey consists in a 5-point semantically anchored scale ranging from 0-4. Average participant ratings on all measured dimensions were between 0.5 and 2.36 indicating a moderate to low burden for a prospective IoT sticker system.

translate the ideas of others into a complete description. The ideas described were usually their own, and they persuaded other members to go along with them.

Summary

Object modifications made demands on family members time, and families differed in how they managed these demands. Some families carved out a specific time of day to work collaboratively. Other families coordinated throughout the day and tasked one family member with implementing the idea. Newer and older families differed in the ways that they desired to realize their ideas. Newer families were aware of new user interfaces and were eager to implement them in their own ideas. Older families had established family routines and hoped that the modified objects could pick up some of the work or help with coordination. As families developed patterns, local leads emerged through working on their own ideas, persuading others to adopt them, or facilitating others' ideas into working modifications.

4.3.4 Potential User Burden

In the survey of user burden, participants thought that an IoT Stickers system presented a low to moderate burden to their household. On average, most of the measured user burden dimensions—privacy, financial costs, and difficulty of use—were rated lower than the scale's midpoint (see Figure 3). These findings are described in more detail below.

Privacy

Participants rated privacy concerns highest of all our measured dimensions of user burden. Participants thought that the privacy policies of commercially available networked devices were somewhat untrustworthy (*M*=2.0, *SD*=1.0). When participants were then asked to envision how a stickers system might enable networking of everyday domestic objects, they were somewhat worried about what information would be shared (*M*=1.72, *SD*=1.2).

Financial Costs

Participants were somewhat concerned with the financial costs of a stickers system. They rated the stickers system as a little expensive (*M*=1.28, *SD*=1.1), and their financial concerns increased slightly when asked about the upfront investment of $300 for an IoT lightbulb (*M*=1.68, *SD*=1.2).

Difficulty of Use

Participants rated a prospective IoT Sticker system's difficulty of use low with the exception of the ability to use it independently. Most participants (n=19) stated that they needed to work with someone else to come up with an object idea very often or sometimes (*M*=2.36, *SD*=1.0). However, once participants were able to come up with an idea, they expended little time (*M*=0.5, *SD*=0.8) and mental effort to create an entry for that object (*M*=1.0, *SD*=0.9). When asked whether the thought process was difficult to get into, participants largely disagreed (*M*=1, *SD*=1.1), and the majority thought that it was only a little difficult or not at all difficult (n=16).

Differentiation

We asked participants to consider all of the objects they had described modifying in their home. They agreed that the system would require them to remember too much information to differentiate one modified object from another only a little bit (*M*=0.84, *SD*=0.7).

Summary

Families raised more concerns about privacy in a prospective IoT Sticker system than all other measured dimensions. While families were somewhat sensitive to the prospective system's cost, the proposed price range was acceptable to most. Members thought that the prospective system was not very difficult to use in terms of the time required and mental effort needed, although they did frequently work with others to do so. On the whole, the user burden ratings did not identify significant burdens with the approach, but did reveal that privacy and independent use would be moderate concerns.

4.4 Discussion

This study uncovered facets of home life involved when families tailor an IoT system to their domestic environment. When families do so, they suspend belief in the current home to consider how objects and rooms might be reconfigured. Object proposals help family members destabilize the working model of home to include previously unthought of arrangements. However, this process of tailoring IoT to the home makes demands of family members' time. This pushes families to adopt differing community dynamics where a local lead fosters that family coordination and collaboration.

Debugging Reconfigured Objects and Roles

To support families in coming to accept a proposal, a tailorable IoT will need to enable collective debugging. Family norms emphasize a need for supporting members with imagining alternative arrangements, and so there is an opportunity for tailorable IoT systems to support experimentation with new roles and trying new configurations. Yet, these systems make demands on family time to ensure all members' views are represented in a proposed modification. To address these needs, the barrier to entry to become a local lead needs to be perceived as low and valuable. Prior work has shown that leadership for smart home installation can be seen as a hobbyist enterprise that is intrinsically enjoyed by one member of the household [49, 151, 232, 35]. However, the findings presented here show that this kind of leadership could be present in only a small fraction of households. Instead, local leadership often happens in coordination with other household members (a finding visible in the results of prior work [151, 35]). There is a shared decision to adopt the ideas of one member or agreement to follow another's lead.

This suggests that lowering barriers to IoT systems should also include making coordination with other members easier through sharing example ideas, eliciting feedback, and prompting others with questions. These kinds of systems that enable groups to work as teams and elect one person as project manager are a longstanding contribution of the CSCW research in the workplace. However, this study identifies an opportunity for translating this system design to at-home collaboration. This is not to suggest that the same workplace values of efficiency and productivity are appropriate, but some domestic contexts are appropriate for using collaborative management tools in order to support social debugging. For example, modification designs that supported voting mechanisms would enable other family members to contribute an opinion about whether the modified object would be successful. Copy/paste mechanisms would allow one person to borrow the idea of another person but extend it as well. Lastly, an annotation mechanism with a prompting question could bring one household member's attention to an incomplete modification initially created by another. Although prior work [35] points out the importance of debugging new DIY-IoT rules during the course of everyday routines, I argue that an important part of debugging will be how relationships are newly configured. Unlike the debugging process discussed in the end user development literature [80, 35], families will need ways to debug newly constructed relationships.

To meet this challenge, focusing on an originator's debugging process is insufficient. Instead, systems will need to support social debugging so that an object's modification can be tolerated by others and acceptable to all.

Modularity in the Home

This study's findings revealed that families use rooms to categorize their objects and activities. Object types are not treated equivalently across rooms (*e.g.* a TV in the living room verses the bedroom). These results show that family members exercise room-level control by regulating what goes in and out of that room (such as P5b disabling internet access in her children's bedroom). These findings suggest that an IoT supportive of local area networks at the range of the room could enable new kinds of interaction techniques and help families manage costs. For example, room-level networks and room-specific parameters for objects could support fine-grained expectations for different parts of the home (an issue identified in early smart home research [67]). Families could make piecemeal investments in IoT—one room or one object at a time—to better manage and shape the impact newly modified objects have on the family (resonating with a smarthome adoption pattern reported in earlier research [67]). These abilities are well within the reach of current IoT research (*e.g.*, IoT light bulb [89]).

Upcycling

This work is motivated by the claim that the costs of IoT adoption need to be considered in design. A tailorable IoT that can be adapted to domestic possessions could address the hidden costs of replacement such as minimizing waste, evolving family members' attachment to their possessions, and reconfiguring entrenched routines. The findings recommend ways for mitigating these costs such as fostering communities supportive of social debugging and supporting granular control over IoT's introduction to the home in system design. Yet this work engages upcycling questions shared by the research community on whether upcycled objects can, indeed, offer greater value than before modification [209]. This study's findings help lay the groundwork for thinking about these questions by revealing opportunities and needs within current household dynamics around upgrading the home to address experimentation with household roles and historical, social legacies (*e.g.,* gender or age) [67, 209]. While prior work suggests that differentiating subgroups of the populations reveal differing values around upcycling [209], our work identifies opportunities to destabilize traditional groupings—such as gender or age—during the upcycling process by experimenting with the home's malleable relationships.

Prior work called attention to questions around "how to imbue an object with potential for imaginative reuse by individuals" [30]. This study found that families already employ coordination patterns that could enable collaborative creativity by building off one another's ideas and experimenting with alternative roles. There is an opportunity for the embeddedness of IoT to scaffold object-oriented creativity through helping reshape object relationships by suggesting experiments for new object roles or role play with the alternative social arrangements afforded through those objects.

4.5 Limitations

While every effort was made to target and recruit households from lower socioeconomic statuses than previously focused on in smarthome research, this study was unable to fully represent those views here. For one reason, the household's data with the lowest SES was excluded because they did not complete the audio diary task. It is unclear why as they were able to complete the home tour and facilitated audio diary task without any problems. Future work should examine whether this study's findings' emphasize the appropriate facets of DIY-IoT for lower SES families.

This study worked with only 10 households in one American city. It specifically tried to target a representative sample for the location where the study was conducted. Working in depth with each family over 7 days allowed the study to capture nuances in how families collaborate when modifying domestic objects that could not be captured in a shorter or more targeted study. Yet, one city and 10 households is not enough to fully characterize how lightweight modification will occur in all households. Specifically, other cultures should be examined. While prior work has examined how DIY-IoT may occur in Europe and East Asia, this leaves a majority of the world untouched. Different cultures will undoubtedly result in alternative family dynamics, and it remains an open question whether these findings would still characterize those alternatives. A larger number of households should also be examined to ensure validity of the framework for other cities and also at scale.

4.6 Conclusion

From work with 10 American families over 7 days, this study uncovered ways that households might tailor IoT to the existing possessions. This work creates a path for upcycling domestic possessions with new IoT services. This study shows how family dynamics could nurture each household member to contribute to an IoT supportive of the family's model of home. It also outlines IoT design opportunities for social debugging and piecemeal investment at the level of the home's objects and rooms. Future work should examine access to DIY-IoT for low SES housheolds and verify this study's findings' applicability to other cultures and communities. In the work presented here, findings show how lightweight modification of the home's everyday objects creates an alternative pattern for IoT adoption that could be made sensitive to the costs of bringing new technology into the home.

5 The Lightweight Modification Framework for Augmenting Everyday Objects

The previous two chapters' findings describe socio-material practices that can be brought to bare on designing an upcycled IoT. Yet, by themselves, they do not provide guidance on how an upcycled IoT should be created. This chapter uses the earlier results to develop a framework for designers and builders to think about upcycling existing objects with cutting edge technology. This work uses the internet-of-things as a case study of how technology adoption could be reimagined to include the routines, habits, and arrangements that households use to sustain artifacts' roles within and as part of their relationships. Below, this chapter describes a framework that makes the work households do to learn, adapt, and personalize objects within the home an explicit part of the upcycling process. The Framework for Lightweight Modification centers four axes of the household's process:

- Object Relationships
- Family Relationships
- Societal Relationships
- Time Use for Evolving Relationships

Unlike many conceptual frameworks, the Lightweight Modification Framework is not a stage-based model in which each facet can be cleanly separated from the others (see Figure 4). Instead, each stands in tension with the others. For example, generating ideas can be constrained by the shared home model as when feasibility and societal norms upheld by the family curtail ideas before they are ever seriously considered. While others have emphasized the importance of families individuating artifacts and weaving them into their household's routines [9, 156, 213, 232, 35], this framework characterizes how upcycling with cutting edge technologies could be situated in the ways families currently make and construct a shared home by focusing attention on four central axes of domestic IoT: object relationships, family relationships, societal relationships, and time use.

5.1 Object Relationships

While Chapter 4's findings describe how families generate ideas for upcycling, object relationships are at the center of this process. Lightweight modification of possessions calls attention to those objects with malleable relationships. For example, objects in particular rooms can be held to higher ideals than their counterparts in other rooms; and so, are targeted for improvement. The intimacy of the bedroom meant that their owners had greater discretion over modifying those objects without having to compromise with others. Thus, these objects were targeted for meeting their owner's standards of comfort and ease. As they are owned by a single owner or intimate couple, they are subject to highly developed conceptions of what comfort and ease should be like. Transitional object relationships, like improvement, reveal ways that owner-object relationships change and create opportunities for lightweight modification to complement and nurture this change.

5.2 Family Relationships

The framework's second axis—family relationships—emphasizes the importance of families reaching an agreement on the household's norms and a shared idea of how the family should live when contesting and reconciling a model of home. A shared model assigns roles to household members around maintenance and care for the home and its objects. In contrast to intimate spaces like bedrooms and bathrooms, the shared nature of the living room and kitchen meant objects were subject to negotiation among residents. Hence, shared objects bore evolving relationships with other household members. Findings from Chapters 3 and 4 showcase how family relationships are implicated in lightweight modification as when family members use collaborative processes like checking in with one another about ideas or testing whether the idea could be feasibly accommodated by others. These processes maintain family relationships and are important for family agreement and awareness that new norms will be shared.

5.3 Societal Relationships

Third in the framework is the set of relationships the home has with society. These relationships were present in values family members placed on the public face of the living room or kitchen. The living room needed to appear aesthetically pleasing to an external observer and meals prepared at home for children needed to be healthy. Societal relationships constrain which modification ideas will be considered by the family and how those ideas will unfold in any particular home by embodying community norms. These relationships also were revealed when discussing external models such as a restaurant, hotel, or other public spaces that could possibly assign new roles to family members. Public places offered family members new ways to stretch what was expected of their domestic role by suggesting alternative modes of care and maintenance for the home's objects and family relationships.

5.4 Time Use for Evolving Relationships

The final framework axis—Time Use—brings attention to the demands an IoT system places on family members. Notably, some families already have established routines that they use to take time to reflect and plan for day to day activities. This suggests that while time is precious, families are willing to use it to try out new ways of accomplishing subtasks for their already complex routines; or, for less ingrained routines, creating any routine at all. This is especially true if there is perceived value around that time. Chapter 4's findings revealed that some families enjoyed a chance to build off one another's creative ideas when working together, but that this could be thought of as a chore if family members struggle to come up with modification ideas. Further, Chapter 4 found that local leads are critical for ensuring the time is set aside and used constructively to integrate a new modification, however

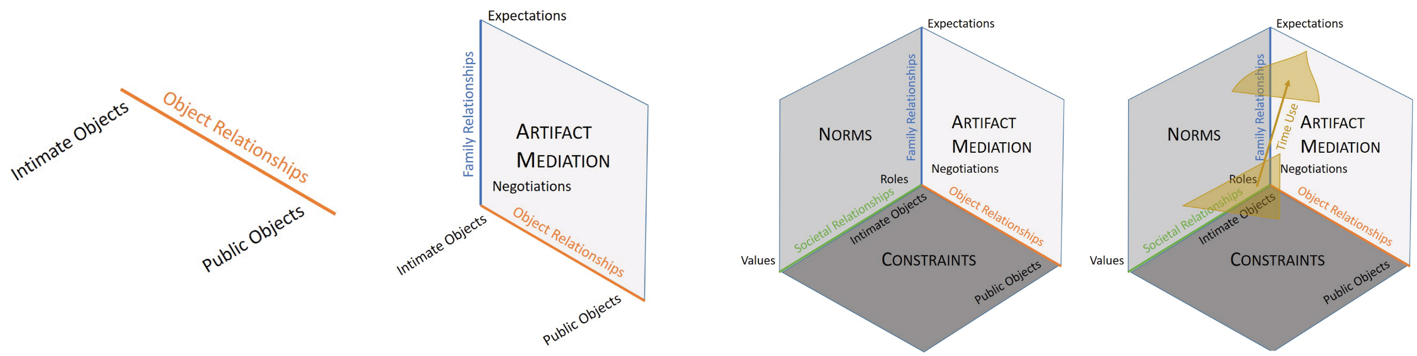


Figure 4: The diagram shows the four axes of the Lightweight Modification Framework. As each axis is introduced, the range of concepts that axis captures are labeled and the attendant interactions are captured by the span of those axes with the exception of Time Use. The way the home's relationships evolve is conveyed as the change of the family's model of the home at two different stages.

lightweight it may be. Time use characterizes the limited resources that families have to invest in evolving object, family, and societal relationships as depicted by Chapter 3's division of labor typologies.

5.5 Employing the Framework

Any design approach that introduces new technology in the home through lightweight modification of everyday objects will need to confront the home's existing relationships and time use to address how those relationships will be reconfigured through modification. Previous work has explored the range of new possibilities modifying everyday objects like scissors, bottles, and coffee tumblers creates [45, 123, 202], but the way these modifications will alter or change the relationships people have with those objects has not been looked at with the same critical lens that social IoT has brought to internet enabled devices [204]. Unlike investment and alteration rituals for previously owned artifacts or disposal practices [174], designing for lightweight modification will require helping owners use their current object relationships and attachments as a raw material.

To show how our framework calls attention to design requirements critical for modification of domestic possessions, this chapter considers an example from prior research on IoT: the Digital Family Portrait [155]. The Digital Family Portrait is a picture frame instrumented with a digital display that updates with data sensed from a remote home. The portrait serves to communicate daily activity information of an elder with their family members living far away. The picture frame is a common household object seen in the living room. As such, the process of observing visual reminders of a loved one conforms to existing object relationships with picture frames. However, the frame alters the social relationships between elderly adults and their family members by disclosing activity information from sensors installed throughout their home. The portrait instantiates new ways of realizing societal values of the family's role in supporting elder members during later life stages by envisioning ways that elderly members can remain at home yet still have access to social support and care through their relatives. These are design features explicitly factored in by the Digital Family Portrait's creators [155].

Additionally, this framework calls attention to trade-offs made during design decisions that need considered going forward. What is the Digital Family Portrait's Time Use? The researchers describe a several month process during which they retrofit the elder member's home with new sensor technologies and tailored the prototype to the field site [192]. While it is unclear whether this is an artifact of the system being a research prototype, prior work on sensor setup and home installation shows that installation is not a trivial task [151, 35]. Leaving this aside, Time Use reveals aspects of how the object and family relationships evolved. The sensor system became invisible to the elderly member and concerns with surveillance by another family member faded as the implications of the system were experienced [155, 192]. Lastly, families developed alternative ways of providing support and caring for one another as they lived with the Digital Family Portrait. Collected data prompted family members to check in with phone calls and prompted new conversations such as inquiring about unusual levels of activity [155, 192]. While monitoring the impact of a new technology on the home is a standard component of user studies, this framework suggests that these findings need compared to an understanding of the same home's coordination patterns prior to intervention to assess the full path of how the Digital Family Portrait evolved relationships and whether such changes were sustained over time.

5.6 Summary

In summary, the Lightweight Modification Framework delimits the principle axes for evolving malleable relationships with domestic IoT. These four axes can inform designers and builders of principle dimensions to address when considering how domestic IoT might be adopted: object, family, and societal relationships, and time use for evolving these relationships. This chapter used the Lightweight Modification Framework to analyze the Digital Family Portrait project. This analysis showed that the framework's Time Use axis reveals the importance of considering the time demands retrofitting a home with new sensors (as typified in smart home research) makes on household life. The next chapter takes up this issue.

6 The IoT Codex: a Book of Programmable Stickers for Authoring and Composing Embedded Computing Applications

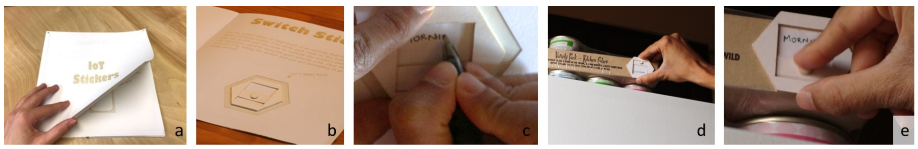


Figure 5: This chapter introduces the IoT Codex: a book of inexpensive, battery-free sensors and interaction patterns to support linking everyday objects to software and web services using stickers To use, a sticker is first found in the the book (a-b), customized (c), peeled from the book and attached to a desired everyday object (d), and then invoked using its kinetic mechanism (e). Stickers can be customized during setup and composed with others to create more expressive applications using the composition space in the book's pages.

6.1 Introduction

The Internet of Things (IoT) promises to extend computing to the everyday objects that make up our physical environment like the home. However, these environments—especially the home—consist in a long tail of random artifacts and the accompanying, idiosyncratic ways that they are used. One of the most popular ways of incorporating this long tail into the IoT ecosystem, is to use attachable identifiers to supply those everyday objects with the wireless communication needed for accessing internet services. To do so, attachable identifiers such as quick response (QR) codes, radio frequency identification (RFID), and Bluetooth low energy (BLE) tags are often stuck on everyday objects so that these passive objects might be a part of a smart environment.

These attachable identifiers are usually passive and so, depend on activity recognition—accomplished through recognizers—to support higher level semantics that can make interacting with them meaningful (*e.g.*, [191, 138, 139, 19]). However, recognizers are ill-suited for the long tail of idiosyncratic, material uses that make up everyday life. Consider the home. Prior chapters showed that households often use objects in ways that are symbolic and contextually situated within small group norms, but these idiosyncratic uses are unlikely to generalize to other households. Yet, state-of-the-art machine learning relies on previous data that is close enough to correctly recognize new use cases. This kind of data is unlikely to accrue in any given household's web of nuanced processes of managing care and maintenance of their possessions; as these idiosyncratic uses often arise in the midst of unpredictable schedules and breakdowns in household routines. Even if appropriate training data for recognizers could be collected, households would need to be able to be able to manipulate the underlying representations of these models in order to effectively engage with them. Yet, recognizers persistently confront issues with their internal representations being intelligible and understandable to users [7].

To support the long tail of socio-material uses of objects, this paper introduces a lightweight approach to customizing IoT services by making attachable identifiers interactive. Unlike recognizers, our approach uses 1) paper engineering techniques to construct attachable identifiers that embody their state, and 2) a tangible, end user programming language to support customizing IoT services to symbolic and situated contexts. We call these interactive identifiers *IoT Stickers*. These stickers enable interaction in a way that reflects decades of research on affordances such as providing intelligibility of system state, supporting user initiation of interaction, and encoding system-supported services with symbolic meaning. IoT Stickers embed a binary tag state in kinetic paper mechanisms. These paper devices provide affordances for invoking system behavior and composing higher level programming abstractions. Thus, IoT Stickers advances research on attachable identifiers by enabling novel interactivity ranging from tangible manipulation to end user programming.

To demonstrate the benefits of making attachable identifiers interactive, we show how IoT Stickers enables customizing IoT services to everyday objects and contexts in a way that can be highly symbolic and situated. Our work contributes

- A set of IoT Stickers constructed using paper engineering techniques
- A tangible, end user programming language facilitated by our system architecture
- Sample applications showing how IoT Stickers, together with our system, supports a lightweight approach to customizing IoT services to suit highly situated contexts
- A workshop study showing the potential of The IoT Codex for facilitating end user programming of domestic IoT

To validate these contributions, this chapter describes 1) the design rationale for IoT Stickers based on iterative and participatory design with families; 2) how paper-engineering techniques imbue attachable identifiers with embodied state through exploded views of fabrication schematics, demonstrations of sticker kinetics, and the corresponding architectural state diagrams; 3) IoT Stickers support end user programming through demonstrating the four affordances successful in blocks programming in a tangible, embodied form: composability, editability, nestability, and geometric arrangeability, and 4) sample applications showing how IoT Stickers supports lightweight customization suitable for the long tail of IoT. The chapter is organized as follows. First, this chapter synthesizes earlier chapters to provide the design rationale for the IoT Codex System. Then, I describe an example IoT Sticker and the underlying system. Next, I describe a set of implemented sticker types, their construction, and how they function. Each sticker is accompanied by an implemented example to illustrate a case of customizing the Sticker. Then, I characterize the ways that IoT Stickers can be customized by further explaining the implemented applications and describing new applications I implemented to illustrate a range of customization possibilities. Since IoT Stickers is a programming language backed user interface, our applications sample a set of possible domestic IoT applications to give a flavor of the design space they enable. Finally, this chapter details a workshop study on designing with the IoT Codex.

6.2 Design Rationale for IoT Stickers

This work used an iterative and participatory process to design IoT Stickers. A participatory process uncovered relevant information about household tasks and elicited feedback from families on how IoT could support domestic life. An iterative prototyping process shaped IoT Stickers to embody end user feedback and task context as well as stimulate appropriate and creative design decisions (in line with [157]).

6.2.1 Sticker Approach

Stickers enable lightweight modification of existing objects by making the process of embedding computing capabilities as simple as sticking them to an the object. Similar to the approach of attachable identifiers, this work uses stickers to enable a household's existing possessions to be incorporated into the IoT ecosystem (for example, see [18]). Each of this work's IoT Stickers associates a unique identifier with both the physical and digital counterparts of an augmented object. For a proof-of-concept implementation, IoT Stickers employs RFID tags. In bulk, these tags can be purchased for <\$0.03. These are passive radio tags capable of communicating a small amount of data——namely, the tag's globally unique ID——when scanned by an RFID reader. In contrast to sensing systems that are integrated in the home's infrastructure and in the background of the user's attention (in line with Weiser's seamlessness vision [225]), IoT Stickers supports explicit end user installation of a wireless RFID network's nodes by enlisting end user choice of what to attach IoT Stickers to and which places are of interest. This sticker approach is likely to impose low user burden in terms of time and mental effort (see Section 4.3.4). The importance of lowering burdens on end user time cannot be overstated (see Chapter 5's discussion of time use), but this does not entail automation. The End User Modification Framework emphasizes how time use and object modification is entangled with family norms, and changing these involves a complex process of family member collaboration and coordination (see Sections 4.3.2 and 4.3.3). Thus, IoT Stickers are designed to minimize time burdens, but not eliminate the collaboration and coordination process of modifying objects with IoT through automation.

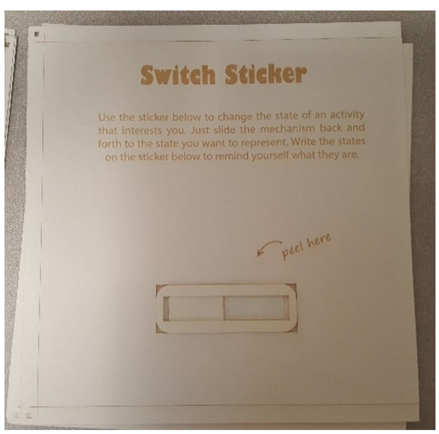
6.2.2 Participatory and Iterative Prototyping of IoT Stickers

IoT Stickers were shaped and designed using an iterative process responsive to the idiosyncratic contexts of American families and their households. The first version of IoT Stickers consisted of 4 sticker types and was envisioned as a smartphone application that used the phone's sensors to interact with stickers placed in the

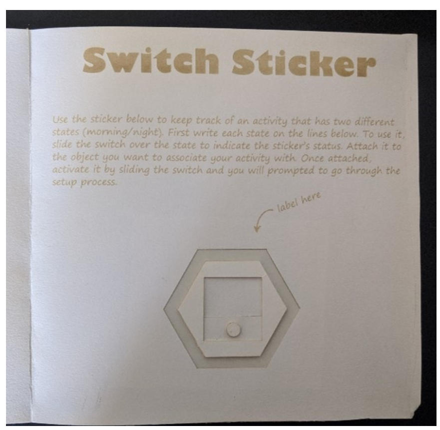
(a) IoT Sticker Buttons from Prototype 1

(b) IoT Sticker Buttons from the Final Prototype

Figure 6: IoT Sticker Buttons were redesigned with pre-programmed patterns that were grounded in everyday household objects.



(a) IoT Sticker Toggle from Prototype 1



(b) IoT Sticker Toggle from Final Prototype

Figure 7: IoT Stickers were redesigned to have a more compact footprint for *in situ* use.

environment. The application adopted a trigger-action architecture in line with other systems for end user programming of IoT in the home (*e.g.*, [3, 80, 218, 219]). This system was redesigned using the following design rationale,

- ***Grounded Patterns for IoT Stickers:*** To help owners think creatively about how they would modify their household environment, stickers were redesigned with pre-programmed patterns that could be remixed. These new designs ground interaction in common household objects to familiarize owners with interaction patterns that could be used out of the box and to encourage thinking about how their existing possessions might be reconfigured. Grounded Patterns emerged from this work's findings that modifying an object cultivates the owner's imagination (see Object Lessons from the Jameson family, Section 3.6.3). Yet, family members typically would work with others to generate ideas for modifying objects with stickers (see Independence ratings in Section 4.3.4). Working with others and the guidance of a local lead (Section 4.3.3), family members often generated ideas through brainstorming and problem solving together. However, defamiliarization helped family members to work independently to come up with an idea (Section 4.3.1). Grounded patterns leverage defamiliarization to cue familiarity with an object, while still suggesting a pre-programmed pattern that might be commonly appropriate for that object (see Figure 6).

- ***Spatial Management:*** As described earlier, families use their control over the home's rooms to nurture their interests and buttress the development of their identity. The family spatially manages household norms by using arrangement to shape expectations about how to behave towards household possessions (See Appendix A.1). The importance of spatial management led to the bipartite analysis of Sections 3.6.3-3.6.4 into *object lessons* and *room lessons*. Two implications for IoT Stickers arose as a result:

  - ***In Situ State Tracking:*** Modifications of shared objects by the family need to support evolving relationships (Section 5.2). Interaction with shared spaces and objects is constrained by families' highly integrated routines as characterized by the division of labor patterns in Section 3.6.2. Stickers were redesigned to have a more compact footprint so that they could more easily be attached to and used with objects *in situ*. This redesign allows for spatial management to have an implicit role in the system's deployment. Since stickers can have a place, then they can likewise carry symbolic, contextual meaning associated with their placement (see Figure 8).

  - ***Room Level Support***: An upcyled IoT system for the home needs to respect the room level norms and control that families exert over modified objects (see Sections 3.6.3-3.6.4). To do this, the RFID reader was moved off a mobile platform to support interaction at the room level using a stationary, room-based hub. For purposes of prototyping, this consisted in a Lenovo laptop running the Windows operating system.

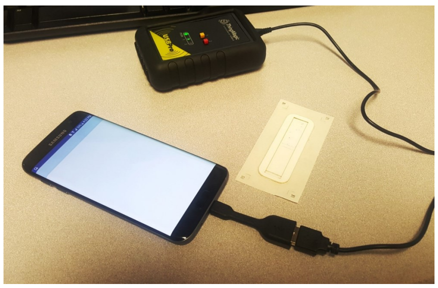
This allows for the reader itself to have a place to support sticker interactions that assume a fixed location with a room-level range of sensing (see Figure 8).

- ***Tangible Composition:*** Families manage home life using a division of labor that accommodates the commitments of its individual members and that structures how they coordinate and collaborate (see Section 3.6.2). These division of labor patterns assign roles and responsibilities to members of the family (similar to specialization that was observed in workplace settings for end user programming [157, 141]). Breakdowns can occur when responsibilities are overly concentrated in one person because others do not always when and how they should contribute. To leverage families' collaborative process, an upcycled IoT should enable setup and maintenance tasks to be subdivided into parallel processes and make each members' role transparent and easy for others to learn. To support this, stickers embody state. This visible state makes the state of the system transparent to users and enables hand-off between multiple family members. Further, the stickers themselves facilitate customization through tangible composition. For example, rather than make a single, monolithic sticker – some sticker states are broken out into distinct hexagonal stickers to facilitate thinking through the process of assigning responsibility for household tasks (this is similar to the editability requirement of Nardi's visual formalisms [157]; See Figure 9b).

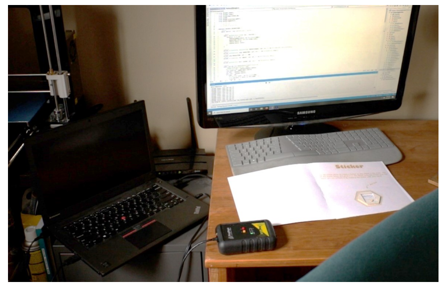
- ***Event-oriented Architecture:*** Trigger-action architectures hide information about triggers' sensed data from the higher level actions that react to them. This hides information about whether the sensed data is uncertain. Instead, event-oriented architectures can pass along uncertainty to those abstractions so that uncertainty can be resolved (see [199, 146]). Many systems attempt to resolve uncertainty by fusing sensed data with other information sourced from other sensors (like microphones or cameras) and through increasingly more powerful machine learning techniques to infer user intent. Considered in the domestic context, this risks normalizing surveillance in a space that should be restorative and can problematically become a question of the system inferring intent within family relationships and division of labor where members set expectations and hold one another accountable. Further, this kind of surveillance is likely not even needed in an end user programming system if the end user is able to directly indicate to the system what to do. IoT Codex adds to previous techniques to resolve uncertain reads by using object classes that correspond to IoT Stickers' embodied state to resolve uncertainty in the sensed data. Thus, end user control over the embodied state of the sticker can indicate what state the sticker's object class should be in.

Summary

Together, the above design considerations translate findings from empirical work with families in the home to shape the IoT Codex. This work used an iterative

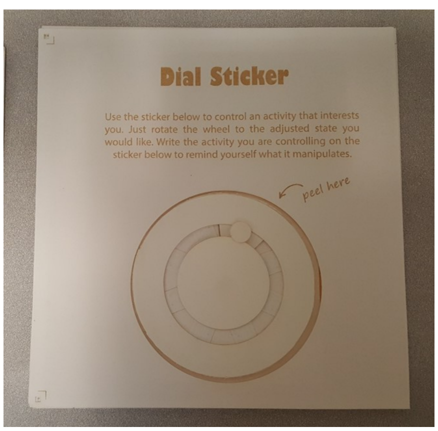


(a) An RFID reader attached to an Android smartphone through USB and Prototype 1 of a toggle sticker.

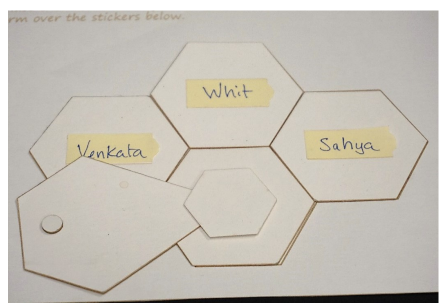


(b) An RFID reader attached to a Lenovo laptop through USB and the final prototype of a toggle sticker in the IoT Codex.

Figure 8: Pictured are versions of two stickers from Prototype 1 of IoT Stickers and an RFID reader attached to a smartphone to serve as a mobile hub.



(a) Early dial sticker from Prototype 1



(b) Dial sticker from the Final Prototype

Figure 9: Stickers were redesigned to support tangible composition.

and participatory process to redesign the IoT Stickers to suit the home. The next sections delve into how these considerations are implemmented in The IoT Codex in much more detail.

6.3 Sticker Fabrication and Operation

In The IoT Codex, stickers are the interaction primitives. These stickers serve as the building blocks for creating a user interface for a user's IoT application. Like links developed for paper user interfaces (PUIs) [93, 143], stickers connect everyday objects in the home to both pages within the Codex and electronic programs and content. To illustrate, this chapter focuses on the button sticker in detail and shares the underlying architecture, before describing the other four current sticker types (Section 5).

I fabricated the button sticker—like all of the stickers presented here—using a layered fabrication approach (see Figure 11). Thus, electronics can be embedded in the sticker in a fabrication-friendly manner. Each sticker variously modifies the basic button layers: sticker paper, double-sided adhesive with a slot for an RFID tag, and a top layer for aesthetic customization like drawing, writing, or icons. RFID tags are used as a proof-of-concept example of cheap, battery-free, wireless electronics capable of embedding digital data in our sticker book without requiring a line of sight as needed for computer vision, optical sensors, or infrared based approaches typical of barcodes or glyphs (*e.g.*, [207, 161, 235]). This point is especially important for this dissertation's application area—the home—where sensing systems that depend on cameras and microphones risk breaching home-dwellers privacy in large part because the computer-recognized semantics are likewise human-recognizable semantics. In contrast, RFID sensing techniques often depend on metadata about tag reads such as received signal strength and channel hopping that do not carry immediately recognizable information such as who said what to whom (see [139] for some common RFID supported semantics). These properties, and the current commercial availability of RFID tags, have motivated their use in the current IoT Codex system implementation. However, it is also important to note that many of the conceptual properties of the system—such as scaffolding of learning needed for end-users to advance to customization and composition of stickers—are independent of this particular technology. These concepts could be adapted to other identification technologies such as bar codes in a different implementation of the interactive book.

RFID for domestic IoT

RFID systems consist of readers and tags. The Codex system uses passive, ultra-high frequency (UHF) tags capable of wireless communication up to a range of 11 meters with an appropriate antenna setup. Prior work has demonstrated that a reader and antenna could be embedded in form factors amenable to easy home deployment. For example, a light bulb equipped with a wifi RFID reader has been demonstrated [89], and in a mass-market setting these could be produced much more inexpensively than current RFID readers. In the future, such advanced form

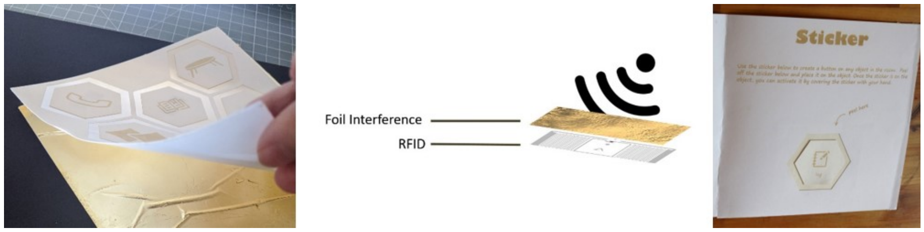


Figure 10: Left: An example fold-out section shows how folded layering enables materials like wax paper and immitation gold-leaf to be embedded in the codex's pages to control whether the sticker is detected by the RFID reader. Middle: RFID is used as an example input technology for the system. At the lowest level, the system interprets input data for RFID tags according to a binary state: uncovered or covered for each tag. The IoT Stickers selectively cover RFID tags with conductive shielding material to make them invisible to the reader hardware by interfering with a read. Right: Button stickers are the first sticker type introduced in the IoT Codex. The page provides initial guidance on how to get started setting a button sticker up. Progressive disclosure about system state and how to advance through the setup process is provided through audio in response to interaction with the button sticker.

factors may be able to facilitate novice installation of a home-scale RFID system. This research builds on this work by focusing on the tags and the interpretation of tag data as an extension of this kind of home-scale setup. Passive RFID tags communicate with the reader when it interrogates the environment for present tags by emitting an RF signal in the 840-960 MHz range. When the tag receives the signal, it communicates with the reader through modulated backscatter—changing how it absorbs or reflects the signal back to the reader in order to encode an ID number unique to the tag. When the tag is covered with conductive material, the reader is unable to communicate with the tag. Thus, individual tags can be represented as having a simple binary state: covered or uncovered. I use this binary state for the stickers (for more advanced forms of tag manipulation see [205, 138]).

IoT Codex Sections

Each sticker type is housed in its own fold-out section. A section uses multiple folds in a large sheet of paper to create standard pages. This folding enables further materials to be embedded into the pages to facilitate interaction. Slots were cut out for the stickers, then layered wax paper and double adhesive were placed around these slots so that stickers could be easily removed and reattached. On a folded page behind the slotted page, there are further layers of double-sided adhesive and imitation gold leaf (ultra-thin copper foil). This layering prevents a sticker from being read by the reader when the book is initially opened (see Figure 10).

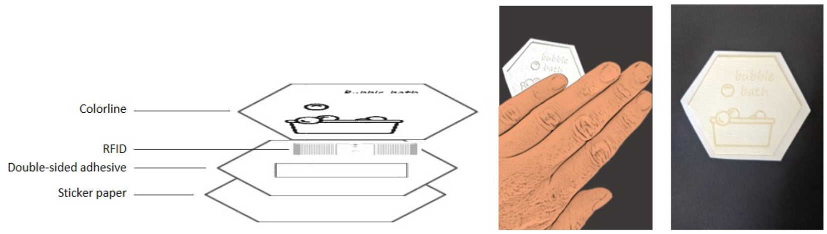


Figure 11: The button sticker shows the most basic sticker interaction: covering the sticker to invoke an audio message.

Button interaction

When IoT Codex is opened to the section for the button sticker (see Figure 11), page text provides initial guidance. It instructs the user to peel the sticker from the page and attach it to the object of their choice to create a sticker button. The first sticker button provides a blank face so that anything can be written or drawn on it. This introduces the user to the logging pattern since this can signal any user-specific meaning and can easily function even with minimal user intervention. This pattern provides immediate value and functionality with little to no mastery of the system's programming capabilities (recent work finds that people derive great value from logging everyday activities using paper-based journals [8, 215, 224]). Once the user first peels the logging, button sticker from the page, this triggers the setup process for the sticker within our system. The logging pattern triggers a brief audio message letting the user know that the sole function of the sticker is to log every time the person presses the button. Once the sticker is attached to the object of the user's choice, it can then be activated by covering the sticker fully with the hand. Once the user covers the sticker with their hand, a nearby audio speaker chimes in recognition of the log. This progressive disclosure provides encouragement and incentive to continue with the button setup process. The steps are also written in the sticker book to serve as a reference if the user has questions.

6.4 IoT Codex Architecture

Events and Event Handling

Similar to most interactive systems, IoT Codex's architecture is event oriented. Its implementation is written in C# and separates hardware specific concerns from the rest of the system through an event queue. Unlike event-oriented architectures for GUIs which deal with semi-standardized input hardware, Codex's system has its own hardware abstraction layer to produce events. The system also enables greater interactivity by supporting manipulation of the parameters to the actions carried out by stickers. This section goes into greater detail about each of these below to characterize the underlying architecture for the IoT Codex that could be extended or adapted to other interactive books, tangible user interfaces, or other embedded computing applications by researchers or programmers. Presentation of the sticker types is left for the next section.

Hardware abstraction layer

I created a hardware abstraction layer that interprets data from input devices providing identification of objects (in this case, RFID tags) to produce events. The system tracks the state of objects visible to a reader and generates events based on the identified object. In Codex's implementation, this layer manages connecting to a ThingMagic M6e RFID reader, and so also the book's physical, sticker devices, and interprets repeated RF interrogations to generate events. Unlike typical input handling—like a key press which generates an event every time a key is pressed—the hardware abstraction layer also generate events when a previously identified object is no longer reported. Since RFID reads can at times be noisy, the reader sometimes fails to read tags that should be visible. Thus, the system provides for interpretation of the incoming signal before emitting events. Prior work has used a Bayesian machine learning model to settle on a decision within 300ms [205]. However, in this system a simpler time-based hysteresis mechanism is used to stabilize reads as in [138]. This approach is implemented using a state machine that tracks the current state of each tag which is seen by the reader and the stability of reads (or absence of reads) from the tag across multiple rounds of basic interrogation by the RFID reader. This process produces the following event types: *Visible*, *Invisible*, *ContinueVisible*, and *ContinueInvisible* When an event is emitted, relevant metadata like RSS and signal phase shift are also passed along to the event queue so that advanced *Stickers* could have the potential to reason about more sophisticated interpretations of the reads (see [139, 138] for examples).

Life of a sticker object

The main component of the system 1. keeps track of the life of a sticker object by responding to the first report of an identified physical object, 2. dispatches events according to a look-up table which maps these identified objects to sticker objects within the system, and 3. uses a state machine and action invocation within those objects as reminiscent of other interactive systems.

When the system encounters the first report of an identified object, it uses a look-up table to determine whether the associated sticker object needs to be created. Note that as will be discussed later, a single physical sticker can be implemented using multiple parts which are identified separately, and which may become visible or invisible at different points in manipulation of the sticker. Sticker objects inside the system are instantiated when the first identified part appears. In order to instantiate sticker objects, all identified parts occurring in the IoT Codex system are pre-registered and linked to a data structure describing the type of sticker object they belong to, and how a sticker object of that type is to be instantiated. When a sticker object is instantiated, it registers the identification information associated with each of its parts in a lookup table. This table is then used to dispatch events to the appropriate sticker object whenever they arrive. Like objects in typical interactive systems, the sticker responds to events dispatched to it by updating its internal state

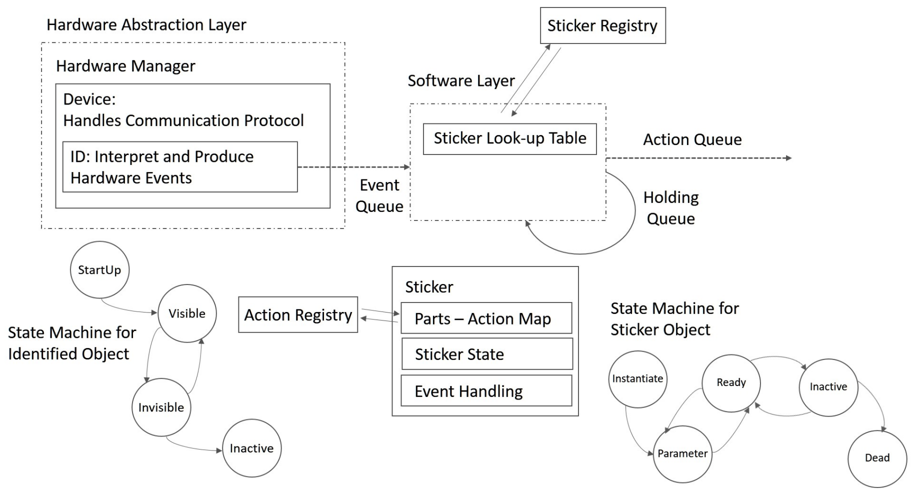


Figure 12: IoT Codex's event architecture to lower the floor for potential end-user programmers of paper user interfaces and IoT applications. IoT Stickers concretize event emitters in a tangible form, and their kinetic mechanisms afford tinkering with parameters and manipulating control flow.

based on its state-machine and/or the internal variables it has access to, and by invoking actions.

Actions

As in most procedural languages, the system establishes a set of parameter values, then passes those to the relevant method/actions. Parameter values can be determined by expressions over literal values, variables (taken from local and global variable spaces), as well as constants, and immutable values established at the time of sticker instantiation. However, the system also offers some unusual ways in which values making up parameters can be established. This includes parameter values which are *asynchronously evaluated*. Parameters of this type include values that need to be obtained externally from services or sensors. This requires asynchronous communication, so the process of gathering an action's parameters must be paused while the system awaits the value. The system also supports parameters that are supplied by the user. To collect these, the system initiates interaction with the user to get the value. Since the system must wait on the user, parameter gathering is again paused while the system waits on the user. This process is similar to the way mediators query the user for help resolving ambiguity in the OOPS system [146, 145]. In addition to asynchronous values, the system also supports values which are established once at the time of their first use, and then reused. This provides an ability for lazy or *just in time* evaluation. This style of value evaluation is typical of user supplied values, because these often impose a cost on the user and it is important to only invoke user interactions if and when they are definitely needed.

Like asynchronous parameter values, actions need to be able to pause in their execution without blocking the full system. To address this issue, the IoT Codex system employs actions in a way that is equivalent to independent threads. We implement this internally by breaking actions into "chunks" and providing a simple scheduling mechanism for these chunks. Chunking provides more flexibility in manipulations of actions. For example, composition operations provided to users support dynamic composition of new capabilities through physical manipulation of stickers (described in more detail in sections 5 and 6). One such case is combining a date-time picker with a play audio file action to schedule playback at the specified time. However, because the UI abstraction provided to users is in only one concrete form – the sticker – and does not separate stickers from the (less concretized) actions they perform, it is necessary in some places for the underlying system to *pull apart* stickers and their actions in unusual ways. Action chunks support scheduling the initial request to the user for a date-time and deferring the later audio playback until the user chosen time.

Summary

The IoT Codex's architecture supports a hardware abstraction layer that 1) produces events, 2) tracks the life of identified objects (in this case, RFID tags), and 3) facilitates interactive manipulation of parameters through advanced parameter types and actions that enable thread-like control. As will be shown further below, these features combine to provide advanced capabilities for the user in a concrete, physically instantiated form, without requiring knowledge/mastery of the abstract programming concepts they embody. Taken together, these features support users with manipulating parameters through proprioceptive control over exposing/blocking tag reads to instantiate stickers and update their state, as well as, directly supplying parameter values through interaction. This creates a ramp for users with no programming skill to develop competence needed to be able to tinker with a programming language. The IoT Codex extends the power of working with abstractions to everyday users by concretizing the architecture's central abstractions of an identifiable object, stickers, and actions in a tangible form.

6.5 Sticker Types and Use Cases

The IoT Codex progressively introduces the system's 5 sticker types through the book's form factor. Page-turning sequences provide an introduction to progressive levels of sticker complexity starting with the Button Sticker and progressing to a Wrapper Sticker. The IoT Codex provides substantial capabilities through fixed form and function Button stickers created for common household tasks. The IoT Codex then gradually introduces customizable Button stickers and eventually progresses to composition of sticker functionality using Wrapper stickers. Each sticker concretizes the link between user interaction and action execution, by explicitly embodying it in the physical sticker's kinetic mechanism. This section introduces the sticker types in greater detail below, and then, immediately following each type is an example use case for IoT in the home.

Button Sticker

The simplest form of sticker supported by the IoT Codex is the Button Sticker (initially introduced in Section 3). As the name suggests, this provides push button style interaction, which can be used to fire an action. A push button can be implemented using an RFID tag by using the user's physical "press" or "touch" action to modify the properties of the antenna it uses for backscatter communication (see Figure 10 and 11). For example, shielding material can be repositioned to attenuate it's overall response, or leave it at baseline (see Figure 10). Alternately, its resonant frequency may be changed to shift with respect to the frequency used by the RFID reader. This can make a previously de-tuned (and hence unreadable) tag readable, or to de-tune a functioning tag to make it unreadable. As described in [138], this can be done in several different ways, including closing (or capacitively coupling across) an electrical contact in order to add or remove an antenna segment, or even by making use of the users themselves as part of the antenna. The sticker book uses this basic pattern of enabling or disabling the ability to read the identification information (in this case from an RFID tag) as the underlying basis for the button sticker and each of the other sticker types described below. As described in the previous section, events reporting these changes in identification status, drive a state machine in each software-side sticker object, which ultimately initiates actions in response to the physical manipulation of the sticker.

Record Message Button As an example use case for the button sticker type, the book contains the *RecordButton*. The *RecordButton* facilitates messaging and communication between remotely distributed people. From formative work with families on upcycling everyday objects, I learned that in one couple the husband makes his wife lunch everyday that she takes to work. The *RecordButton* could be used by him to leave a message on his wife's lunch for her to find and invoke when she opens her lunch at work midday. When this sticker is unpeeled and attached to an object, it prompts the user to record a message using a first use parameter. The user can do so by covering the button sticker to begin recording and uncovering to stop recording. This recording serves as the parameter to the button's newly bound play action. The user can then invoke the action by quickly covering and uncovering the button.

Toggle Sticker

The *Toggle Sticker* uses a slider mechanism to selectively interfere with the reading of one or the other of two tags. The design of the mechanism ensures that the tags operate as mutually exclusive of one another (see Figure 13), and this gives the sticker the capability of functioning as a simple user controlled conditional.

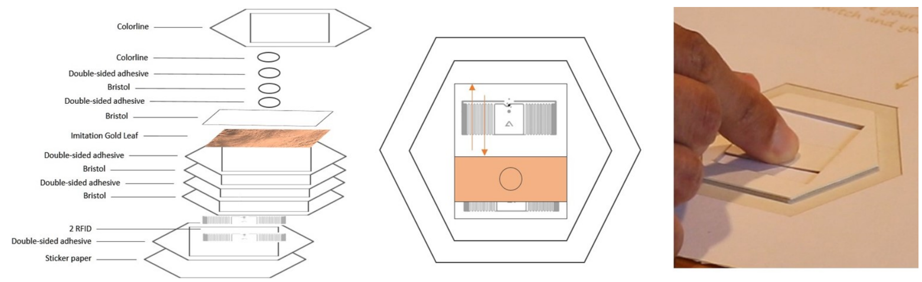


Figure 13: The Toggle Sticker shows an example of a slider mechanism used to selectively activate or deactivate an RFID tag. The orange parts of the diagram show how the imitation gold leaf has been embedded into the layered fabrication of the sticker and how the kinetic mechanism moves the shielding to cover or uncover the tag. Since the cover can only cover one of the two tags at a time, it effectively enables the user to set up small conditional constructs. Ours shown in Figure 5 logs feeding the cat for either the morning or the evening.

Cat Feeding Toggle The scenario, described in the introduction, depicts a case where a *Toggle Sticker* could be customized to address situated needs in the home. In the scenario, Judy uses the sticker to coordinate with her housemate on whether or not the cat has been fed. Using the sticker, Judy creates a situated message on the cat food container for coordination purposes that is able to simultaneously record the state of food supplies within the house. This can help Judy with knowing when she needs to purchase more cat food even in cases when she hasn't been the one to recently feed the cat, and so did not directly observe the food depletion.

List Sticker

The list sticker is implemented using 3 RFID tags in a row covered by pop-up style flaps. Each contains shielding which can be lifted away for the tag (see Figure 14). This supports a user with specifying a choice within a fixed set of values and allowing for greater complexity in the relationship between those tags if the user desires. Since more than one RFID tag can be activated at a time with the list sticker, more complex situations can be managed with this tag. For example, partially fulfilled tasks could be managed and tracked with the list sticker like a To Do list.

Bedtime Checklist The *BedtimeList* was created in response to parents who said that they would like to scaffold their children taking control over important routines without them—the parents—having to remind and nag them. The bedtime list was created for a scenario in which a parent needs to be away from home when the child goes to sleep to show how the list sticker can provide this scaffolding, but also facilitate expressions of care even when family are remote from one another. The list contains actions that 1. plays an audiobook for the children's story *The Carpenter and the Walrus*, 2. plays a two minute song to remind and accustom the

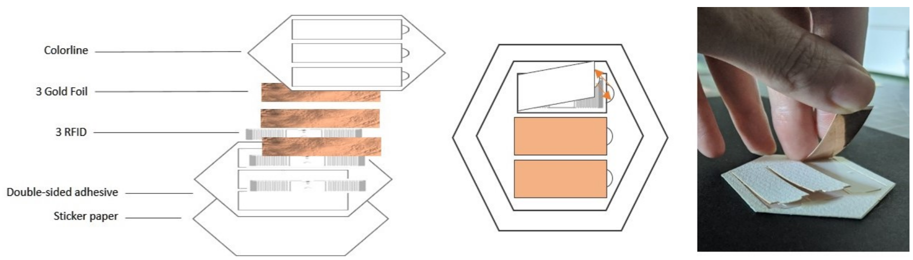


Figure 14: The list sticker is implemented with multiple RFID tags, that can each be activated/deactivated by opening and closing the flaps. The list sticker allows for actions to act in combination with one another or for a simple set of items to be collected together. Above, we show a checklist in which items are 'checked off' by opening the flap, and scaffolds a bedtime routine for young children.

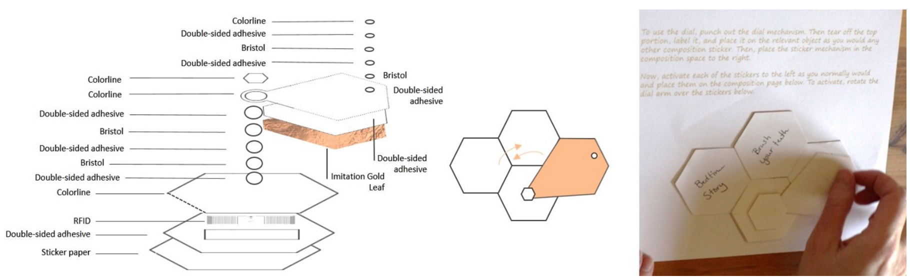


Figure 15: The dial sticker is pictured showing how a menu could be created using stickers by adding a procedure for each sticker item that could be selected when the dial arm is rotated.

child to brush their teeth for a full two minutes, and 3. a message of love, saying good night from the parent to the child. Although the parent is away, the child can use the list sticker to go through their bedtime routine at the appropriate time, and the parent can review it and rest assured that everything is as it should be. Even in cases when the child is likely to have an alternative caregiver around, the sticker still provides value by modelling fluent reading for the child, practicing hygiene behavior, and a way for the parent to say goodnight. The sticker supports the parent with scaffolding this routine even when they cannot be present to do these in person.

Dial Sticker

The dial sticker is implemented using 6 RFID tags fanned out in a circle and uses a rotating lever mechanism. Each tag can be deactivated when covered with the dial's moving arm lined with foil. The dial sticker supports interactions that require a selection between a small set of values or states. This could include controlling the place of play in a video/audio file, manipulating rotation of a 3D object, or scrolling. Dial stickers are also supportive of cases where alternatives describe social relationships rather than the system's objects, such as identifying who in a set of people is responsible for taking some action like doing a household chore.

Playback Dial The *Playback Dial* was created for providing greater control over audiobook playback. This example uses 4 of the available sticker slots to support 1. play, 2. pause, 3. unpause, and 4. stop. Since social reading rarely proceeds as a linear activity where reading proceeds from start to completion, the *Playback Dial* allows for greater control over when and how the audiobook supplements reading the physical book's text. This allows for the sticker's functionality to support reading with audio on demand. Moreover, since stickers offer a cheap form factor, this functionality could be duplicated for many books to encourage development of visual literacy skills.

Wrapper Sticker

The wrapper sticker consists of a larger shaped sticker that contains a slot big enough for a sticker to fit in it. A wrapper modifies the action of the sticker inside of it in a simple, but useful and generic way. This provides a concrete means to compose and manipulate sticker behavior. For example, a wrapper that asks the user for a date and time could support 1. *Delay*, waiting for some period of time before carrying out the action or, 2. *OnlyOncePerDay*, blocking subsequent action invocations until the next day, but then allowing the next one to occur. Overall, wrapper stickers provide small modifications to existing sticker functionality through simple compositions of sticker functions, and do so in a concrete and physical form.

Spa Time Wrapper The *SpaTimeWrapper* was created in response to users desiring a way to create temporary, leisure spaces in the home. The wrapper schedules the spa settings sticker to run on the date and at the time that the user plans to take over the bathroom for a bubble bath. The wrapper attaches to a paper door hanger that the user can punch out of the book and that reads "Do Not Disturb". The spa setting sticker plays a list of relaxing music. This wrapper sticker provides a situated, physical message to others in the home who might approach the bathroom through the door hanger, but also coordinates IoT appliances to create a spa scene when the user wants to take a bubble bath.

Summary

The current IoT Codex supports five sticker types for a customizable IoT user interface. To show how the *Stickers* could be tailored to domestic IoT use cases that are situated and idiosyncratic, the implemented example stickers are parameterized through *Action* parameters or the *Actions* themselves to suit highly specific needs as required by the described scenarios. These examples illustrate how the IoT Codex uses kinetic mechanisms to gradually introduce more complex, embedded computing capabilities by leveraging the sequential nature of the book form factor and by

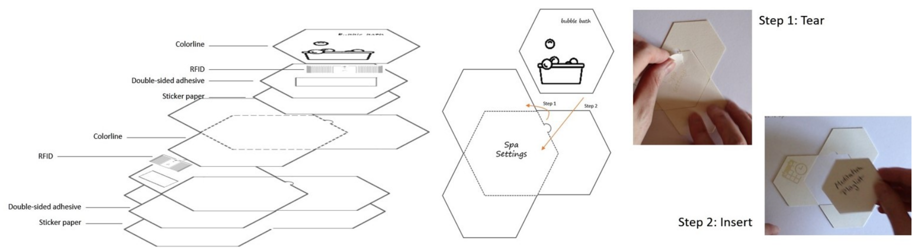


Figure 16: The wrapper sticker shows how stickers can be composed together to customize more complex behavior. The two photos on the right show the two step process: 1. tear out the center shape, and 2. insert another sticker into the middle hole.

encouraging manipulation and tinkering with the stickers themselves. By concretizing the association between user actions in the physical world and computational actions by means of Stickers, the IoT Codex exposes user to tinkering with low-level details without requiring mastery of low-level knowledge or programming.

6.6 Customization

The IoT Codex system seeks to lower the floor for potential end-user programmers of paper user interfaces by gradually introducing customization. By concretizing programming abstractions as tangible stickers, IoT Stickers facilitate tinkering and tailoring embedded computing applications through increasingly powerful means including: labeling and contextualization, kinetic paper mechanisms, sticker juxtaposition, and combining them using special composition stickers—here, called *wrappers*—to compose more complex behavior. Below, this section describes in greater detail how the IoT Codex supports these techniques for progressively introducing tailoring skills through supported customization mechanisms.

Labeling and contextualization

The IoT Codex and Stickers use kinetic mechanisms from paper engineering and the physical properties of paper and stickers to first introduce users to customization. At the most basic level, stickers encourage customization through writing or drawing on them as with other PUIs. This is first introduced in the book as switching from a pre-programmed sticker (like a logging sticker) with an engraved icon to one with a blank face on it. The pre-programmed behavior is the same, but when the user peels off the sticker to enable this behavior, they can attach their own semantic meaning to it through annotation. Because stickers are essentially made from paper, they leverage the user's familiarity with writing and paper craft to make customization approachable through lightweight and inexpensive form. Similarly, stickers can be easily contextualized through placement. It is at the users discretion where the sticker goes and what it attaches to. This allows for the user to leverage their knowledge of social context and ritual usages of certain places and locations to customize the sticker.

Kinetic manipulation

Other kinds of kinetic manipulation are also important for the system. The IoT Codex system uses the book form factor to progressively introduce programming concepts. The book progresses from simple peel-and-stick customization to wrapper based customization. This sequence allows for progressive disclosure of more complex ways to customize the stickers. Earlier stickers—buttons, switches, and lists—use a single sticker's kinetic mechanisms to invite tinkering and experimenting with that sticker's behavior. When the mechanism is triggered, first-use behavior prompts the user to supply parameters to customize its behavior. For example, one of the basic actions is a text-to-speech action. When triggered, the system shows a dialogue box asking the user to supply a string to be spoken by the speech synthesizer. This parameter is stored by the system so that subsequent invocations of the sticker will cause the speech synthesizer to speak that text. This leverages form-filling to acquaint users with customizing parameters (a similar technique is used for query formation in [171]). The *Record Button* shows another example of this same first use behavior, but eliminates the need for a keyboard and screen by allowing the interaction to unfold in an audio-only format that can readily supplement the tangible interaction of the book and stickers. This principle lowers the floor for tinkering with a programming language's parameters by changing the parameter's values.

Juxtaposition

A user can begin to compose behavior in more sophisticated ways through physical juxtaposition. Placing two otherwise independent stickers in physical proximity to one another in context can allow for greater functionality. Pre-programmed stickers such as logging stickers or play-an-audio-file stickers can be physically placed next to one another to allow for their behavior to be combined. For example, a set of stickers set up to play single notes can be placed next to one another to allow for a piano keyboard of sorts to be created. This could allow for playing a song or even turning everyday objects into music-making devices. Similarly, a set of logging stickers could be set up to log different activities to create a self-tracking application. For example, logging stickers set up to log time spent on work, chores, and play could allow someone to track where their time goes and what they spend their time on the most. In sticker form, this enables these activities to be tracked *in situ* using the objects most associated with them such as a laptop, refrigerator, and gaming console. As needs evolve, the sticker form factor enables easy removal and replacement with different IoT behavior. For example, as a person discovers that they frequently work past their intended stop-time, they could replace a logging sticker with an alarm sticker that schedules a chime at the stop-time to remind them to switch their activity. This could help them uncover whether stopping at their intended time gives them enough time to do all their household chores and still take a break at the end of the day.

This work experimented with two different techniques to introduce users to combining stickers together when setting up the dial sticker. For the first technique, printed sticker outlines were created on one of the dial section's pages to help users align the appropriate sticker with the appropriate slot. Instructions were printed on the page instructing the user on what order to sequence placement and provided labels within the printed outlines. Activating the dial was made even more explicit through interaction with the book to reify the idea of instantiating behavior: the dial arm must be punched out of the page along the designated perforations while its composing stickers are unpeeled like the simpler stickers. The second technique used for the dial leverages the page-turning behavior of books to sequence placement of the stickers. The composition page (the page printed with sticker outlines) is overlaid with a semi-transparent page with holes punched out in the shape of simple stickers for the places where the dial's composing stickers should be placed (this was done with only four instead of all 6 composing stickers). This guides the user to place the simple stickers first before placing the dial arm since the simple stickers can pass through the holes of the overlaying sheet, but the dial arm cannot.

Composition through Wrapper stickers

The Codex form factor, in addition to the stickers' kinetic mechanisms, introduce composing stickers together to create truly new behaviors. Like the dial sticker, holes, tearing, and shape cue composition of the wrapper sticker with another, simpler sticker. The wrapper sticker itself has a perforated middle hexagon that must be torn out of the center to create a hole for the wrappee (the simpler, smaller sticker) to go in. Further, the wrapper is made intentionally bigger—the size of three haxagonal stickers together—to cue its "wrapping" behavior by making it big enough to hold another sticker. The wrapper also uses a printed outline on a composition page to encourage sticking the wrapper to the page first before sticking the wrappee.

6.7 Design Space Validation

A workshop study with 3 participants was conducted to validate the IoT Codex's design space.

6.7.1 Method

The study lasted 2 hours and took place in a university setting. Participants were compensated $15/hr. for their time. Study tasks consisted in completing a background survey, watching an IoT Codex Demo video and handling the physical prototypes, designing applications for the IoT Codex, and completing survey questions about their experience designing with the IoT Codex.

Participants

Previous research shows that end user programming is a collaborative activity that involves multiple people with varying specializations [157, 141]. This research has shown that relatively advanced users help relatively less advanced users through their knowledge of program features [157]. Within this community, local developers provide critical support by bridging domain expertise with developer expertise [157]. To validate the IoT Codex's design space, the workshop study recruited participants who could fulfill this local developer role by bridging both domain expertise through knowledge of their own homes and household dynamics with the expertise needed to understand how to design and program interactive systems. Thus, participants were recruited through listservs and internal messaging channels that served specialists in human-computer interaction. Participants were screened using a 2 minute survey that verified whether the participant had taken core courses in human-computer interaction or who could demonstrate equivalent experience.

Procedure

The study's procedure focused primarily on creating applications for the IoT Codex. When participants first came into the lab, the researcher went over the consent form and answered any questions. Then, participants were asked to complete a background survey covering their demographics and experience with creating interactive systems. Next, they were shown the demo video of the IoT Codex. Then, the researcher showed the participants IoT Codex's physical prototypes and introduced each sticker by describing one of the demo applications built for that sticker. Throughout, participants were given chances to manipulate the IoT Stickers and ask questions of the researcher. Following the introduction of the IoT Codex, participants completed 3 design sessions focused on creating applications for 1.) pre-programmed IoT Sticker designs, 2.) first-use parameter IoT Sticker designs, and 3.) composing IoT Stickers together. Finally, participants completed a survey rating their experience of designing applications for IoT Stickers.

Data and Analysis

The study session was video recorded using two cameras set up around the shared table so that all participants could be recorded. Audio recordings were transcribed by a professional transcription company. The written transcripts were then coded for themes of interest [28].

6.7.2 Findings

Participants

Participants ranged between 21-26yrs old. Two participants identified as female and one identified as male. All had a college level education, and one had a master's degree. Two described reported having 1-2 years of experience designing interactive systems, and the other reported 3-5 years of experience.

Customizing IoT Stickers to Participants Homes

Participants envisioned controlling which people, objects, and tasks IoT Stickers were assigned to. Attaching IoT Stickers to an object gave it context. For example, P2 explained, *"to be contextual to the idea I was sharing about, I'd probably just stick it to the door or somewhere in the entrance."* Two participants lived with roommates, and they envisioned assigning IoT Stickers to the person responsible for doing a task. For example, P3 wanted to use toggle stickers to track whether or not roommates did their chores. One participant who lived alone wanted to use the system to track the state of half finished tasks like laundry so that she could remember where she left off. She explained that living alone meant that she couldn't rely on housemates to help make her aware of what needed her attention, and she thought the system could help with this. Giving IoT Stickers context attached sticker support to the relevant people, objects, and tasks of participants ideas.

Participants all thought that IoT Stickers could help with interpersonal communication and coordination around household objects and tasks to enhance their relationships with others. P1 remarked, *"I feel like it fosters good communication. It can help maybe avoid, like in a house situation, avoid passive-aggressiveness"*. They were happy to offload household management that everyone agreed to on the system rather than carry the burden of it themselves. She elaborated, *"You can have the automated...'bing,' like, 'system has notified you that it's your turn to do the dishes'"* (P1). Offloading this reminder to the system did not mean that participants wanted the system to automate reminders. Instead, they wanted to adopt the anonymity afforded by the system. For example, P3 described linking a button sticker with the roommate's name to a toggle sticker tracking whether they did their chore. Then, 3 button-toggle combos for each of 3 roommates could be displayed in a column. P3 explained how a person could see the column of chores and recognize who hadn't done theirs. Then they could press the button sticker for that person so that the system would notify them. In addition to household chores, participants also envisioned customizing IoT Stickers for celebrations and aspirations such collaboratively planning a movie night or adopting a new skin-care routine. In general, the IoT Stickers were envisioned as a way to manage communication about how the household should run without making it a personal task to nag someone else.

Concretizing Applications

Tangible initialization of first-use parameters offered a way to concretize application ideas and customize them to context. First-user parameters were used to customize which people, objects, and tasks IoT Stickers were assigned to. P3 explained to the group, *"So maybe fill out this and then attach it to the washer or dryer. It would need to be parameterized with what the object is."* The ability to customize the IoT Stickers through tangible mechanism helped participants envision program sequences and timings: *"there's someone's birthday coming up, and I think when you added this complexity, it helped me concretize with that idea."* P2 admitted that he had a hard time coming up with ideas for first use parameters, but he thought

IoT Codex's structure helped with gradual introduction of programming ideas. P2 explained how *"I think it's kind of like, say, teaching a programming language or something like that. We start with the simplest feature that this offers you and give a very concrete example. And then after introducing all of those through the booklet, people can get more creative and be more flexible."* The tangible and kinetic features of IoT Stickers supported specifying the relevant context.

Collaborative Tangible Composition

IoT Stickers' tangibility coordinated multiple people working together. P2 highlighted how IoT Stickers' embodied state minimized digital notifications: *"we wouldn't need additional notification or something, because I think the advantage of this digital, kinetic device, is that you see what's done there already."* IoT Stickers' support for coordination included learning from one another on how to translate ideas into sticker applications. P2 exclaimed, *"I think P3's idea was really brilliant, on using this to indicate two people."* The ability to tangibly reference an IoT Sticker supported participants with illustrating their concept. P1 built off of P2's idea by taking the stickers P2 had been using and arranging them on the table to illustrate and ask a clarifying question. Similarly, when P2 described composing two stickers together, he picked up the dial sticker and rotated the arm to show his idea. IoT Stickers' embodied context supported participants with sharing the same programming idea. Participants thought this feature would help coordinate members of their household. P3 explained how *"the people closest to the control do the thing and make it transparent across everyone else."* IoT Stickers' embodied state provided a shared reference that supported communication and coordination.

6.8 Discussion

This chapter showed how the IoT Codex could support a lightweight approach to programming embedded computing applications. By concretizing identifiable objects, stickers, and actions, the IoT Codex scaffolds exposure to low-level details of programming IoT applications. However, the book of inexpensive, battery-free stickers can be used to create a UI without knowledge or mastery of any of those details. As a paper interface, it is possible to customize the stickers through annotation, parameter manipulation, juxtaposition, and composition allowing for users to tailor embedded computing applications. Further, these programmable stickers enable users to tinker with parameters and actions as a way of re-mixing pre-programmed design patterns to encourage customizing applications to idiosyncratic needs. Below, this section discuss some of the possibilities for future work that this research enables as well as some limitations to what has been shown here.

Identifiable tags

The IoT Codex integrates advances made in PUIs and TUIs by employing RFID as an example input device. RFID supports the rich affordances of the kinetic paper mechanisms that matter for concretizing programming. However, the Codex's system architecture could support other kinds of input devices that can supply a unique ID. Barcodes and computer vision techniques are obvious extensions of this work (see [119]). However, recent work on cheap, battery-free, wireless sensors opens up wider opportunities for looking at other kinds of input devices for the architecture: tags that are radio-based [237], triboelectric-nanogenerator-powered [15], or textile-embedded [125]. This work proposed a book form factor for introducing parameters, actions, and composition techniques to progressively develop expertise at creating embedded computing applications. If these tags could be integrated with paper (like the triboelectric-nanogenerator of [111]), this would open up a wider range for kinetic mechanisms for interactive books. Another fruitful line of work would be to investigate kinetic mechanisms for barcodes in interactive books. Prior work has shown proof-of-concept, kinetic paper-based devices ([239]), but their programmability and sticker properties have not been examined. This integration would enable interactive books to achieve an even lighter-weight deployment through paper and phones, but would require a line of sight. Even now, this capability is not out of the reach for the current system, as smartphones often are equipped with NFC readers. These are near-range equivalents of RFID, but have a much smaller read range of centimeters in contrast to the UHF RFID tags's range of 11m.

Abstraction and exposure to low level details

The IoT Codex focuses on supporting users with the lowest tailoring skills. Received wisdom characterizes the people with skills to tailor a system to suit their idiosyncratic needs as lying along a spectrum from those focused on everyday tasks and activities, to those focused on programming and creating extensible systems [144]. Tinkerers are situated in the middle of that spectrum and can manipulate parameters to the system's programming language, but have no mastery of that language. Likewise, people involved in everyday tasks, have the lowest tailoring skill, and so the least power to tailor a system to their wishes [144]. Recent smart home research echoes this spectrum with a range between pilot and passenger users [122]. Yet, within the category of pilot users the research does not distinguish between tinkerers and programmers [122]. In this work, the IoT Codex scaffolds tailoring skills without needing to master standard languages like Java or visual languages like Adobe Acrobat form design. Specifically, The IoT Codex supports tinkering with parameters and remixing pre-programmed patterns (see [54] for a discussion of the role of re-mixing in developing competency).

The IoT Codex uses kinetic mechanisms to balance abstracting lower level details to support high level composition with exposing users to those same details in order to scaffold introduction to programming concepts. Stickers provide a basic set of interactive objects that serve as the architecture's primary abstraction so that the user can quickly and easily compose an interface. These abstractions serve as an important contribution of toolkit research. Yet, by hiding these details, these very same abstractions can create barriers for novices because they hide the information needed to develop competence and expertise [150]. This has led to researchers calling for *untoolkits* as a contribution. Untoolkits thoughtfully design basic building blocks so that novices can use and be exposed to the same tools experts use. The IoT Codex bridges a need for valuable abstractions with the need to scaffold exposure to low-level details to facilitate new groups of people being able to use the same components and materials used by experts. However, it stops short of contributing a full untoolkit, by recognizing that users can derive significant value from these abstractions through tinkering with parameters and compositions. Importantly, for this dissertation's use case, the home, users may not want to devote the time and effort needed to develop programming expertise, but still need a way to customize and exert control over smart home systems [56, 122, 157]. The IoT Codex supports them without requiring they learn to become a full fledged computer programmer.

Limitations

This work proposed the IoT Codex for smart home and ubiquitous computing applications. However, it only assessed the Codex's architecture with a small set of tags at a time (<10) since many of the supported interactions are concentrated in and around the book. If the stickers were to be truly ubiquitous and placed on many household items, the system would need to handle different materials and form factors. Future work should evaluate the system with tags placed on a variety of household objects to develop an understanding of how tags may behave differently when attached to differing materials. Further, each section of the book was tested in isolation from one another. A technical evaluation should be conducted to understand the challenges that may arise when the book is collated. Along these lines, this work has not yet tested the IoT Codex's usability with users. Although the stickers provide affordances that have been well tested by the pop-up book industry, user manipulation in an ecologically valid setting may introduce uncertain read data or false classifications for the system. A usability test would uncover these.

6.9 Conclusion

This chapter presented the IoT Codex system to support authoring and composing embedded computing applications for IoT. The chapter described how paper engineering techniques and the book form factor could be used to create alternative abstractions and interaction techniques for end-user programming. This work contributes a book of programmable stickers that convey the affordances of an EUP language through stickers. These mechanisms constrain what expressions can be composed in the language by using both shape and kinematics. In doing so, the resulting IoT Codex contributes physical computing techniques to aid selecting appropriate EUP elements and remixing them to program IoT applications. The event architecture supports lightweight association between wireless, battery-free sensors and IoT services. Specifically, it supports tinkering with and remixing preprogrammed behavior to develop tailoring skills as needed for adapting IoT to the home. This chapter showed how the IoT Codex's form-factor and architecture could support domestic scenarios of tailoring IoT to the home's idiosyncratic needs. This work demonstrates a way to associate IoT services with a dramatically wider set of objects and tasks than previously supported.

7 Contributions and Open Questions

A central challenge for end user programming in the home is developing a system with the procedural flexibility to adapt to the idiosyncratic context of the home. This difficulty is exacerbated when we begin to ask whose home and which context? Even if the question is confined to the American home or even that of 10 families in Pittsburgh, clutter can readily be observed. Clutter highlights how messy real homes are and how difficult it is to generalize across household so that some trends will prevail. It is far too easy to dismiss this clutter and design for some idealized family who never makes a mess. Yet, if we take this clutter seriously, we can see that it has a central organizational role in the home.

If we abandon creating systems for idealized families and instead, create systems suitable for their idiosyncracies, then we need systems that can adapt to their clutter. Heavy-weight approaches, like context-aware computing, have sought to infer context so that smart homes might adapt to their users. Yet, by trying to infer context, these systems tend to employ always-on ubiquitous sensing to monitor their users. This approach normalizes surveillance in the home and might ultimately never be able to appropriately infer the intent of clutter. Often the use of clutter, and many other objects in the home, carries symbolic meaning. Inferring this symbolism will be incredibly challenging especially if families are prone to changing the rules and norms that shape daily life so that their home might evolve. Instead, we need a lightweight, cheap, and customizable IoT that can accommodate daily clutter.

This work showed how to create a lightweight, cheap, and customizable IoT appropriate for the American home. Because the American home is filled with daily clutter, this dissertation adopted an upcycled approach to IoT so that such possessions could be part of the IoT ecosystem. To develop an Upcycled IoT, this dissertation uncovered

- Ways that socio-material practices are implicated by and could be reconfigured when domestic possessions are augmented by IoT
- Techniques enabling existing possessions to be transformed with IoT services
- A system to enable end-users to customize IoT to suit their domestic context

Taken together, these contributions show how an upcycled IoT could support installing, customizing, and managing the introduction of new IoT capabilities into the home in the form of the IoT Codex.

While this dissertation's contributions show how tangible end user programming offers flexibility in specifying context for domestic IoT systems, open questions remain.

7.1 Spatial End User Programming

If small groups are to have meaningful control over IoT services, we need new system architectures. I uncovered the symbolic significance families attach to their household's spatial layout and use spatial configuration to regulate household norms.

Future work should investigate how distributing end user programming controls throughout a space facilitates programming shared IoT services. For example, IoT Stickers' architecture could be extended to provide programmable abstractions that enable classifying and manipulating collections of stickers with symbolic, spatial meaning like all of the living room stickers. The ethno-archaelogical methods that I adapted for this dissertation could be used to uncover how well these abstractions align with a family's understanding of spatially manipulating and shaping norms. Extending IoT Stickers' architecture and replicating this dissertation's methods could open up new forms of interaction that better align with families ways of organizing home life.

7.2 Adapting to Everyday Objects

This dissertation charted a path to include everyday household objects into the IoT ecosystem. While IoT Codex offers a set of stickers that are designed to accommodate many object forms of household objects, there will be many others that it is ill-suited to. Part of this limitation is the size of the sticker itself. It will be unable to adapt to small ledges, odd corners, and the various other nooks and crannies that make up a home. Another part of this is due to the underlying technology being made up of RFID. As explained in the construction of the IoT Codex, metal shielding will interfere with RFID reads. Yet, many artifacts in the home consist of metal material. Other material interferences remain unknown. Future work should examine how viable RFID would be as a sticker based technology in an ecologically valid setting. I prototyped stickers that use barcodes to begin to address these limitations of RFID, but did not redesign the size of the sticker. A more thorough characterization of the material idiosyncracies of the home is needed to begin to make progress on a system that could augment these everyday objects in a similarly lightweight and tangible fashion.

7.3 Tangible End User Programming

This dissertation showed how the IoT Codex could support end user programming ideas for the home by developing a proof-of-concept system and running a workshop with human-computer interaction specialists. However, a more thorough understanding of how IoT Codex does or doesn't support tangible end user programming would require an more extensive study in which end users have a chance to refine their ideas using the system over a longer period of time and concentrated on fleshing out an idea more completely. When the IoT Codex is employed to work out a full fledged application for the home, the fitness of the IoT Stickers for the task can be better uncovered. Such a study may discover unmet end user needs that challenge the current design and architecture.

7.4 Conclusion

This dissertation worked with 10 American families to design an upcycled approach to IoT that makes use of existing household possessions and then built a system responsive to these findings. The results 1) describe patterns of families' socio-material practices, 2) develop a framework for designing lightweight modification, and 3) presents The IoT Codex—a book of programmable and inexpensive, battery-free interactive devices—to support customizing everyday objects with software and web services using stickers. The presented work offers a lightweight approach to end user programming of everyday objects for customizing IoT to suit the idiosyncratic socio-material practices of the American household.

A Appendices

A.1 Spatial Management and Floor Plans

Families use their control over the home's rooms to nurture their interests. They use spatial management to buttress the development of their identity. Below, Figure 17 shows how bedrooms allow the owner to exert more control. The color coding on this floor plan reflects the 3 participants' use of artifacts and technology in the home. Royal blue is the color for all shared objects, but pink, yellow, and turquoise shows individually owned objects. The family spatially manages household norms by using arrangement to shape expectations about how to behave towards household possessions.

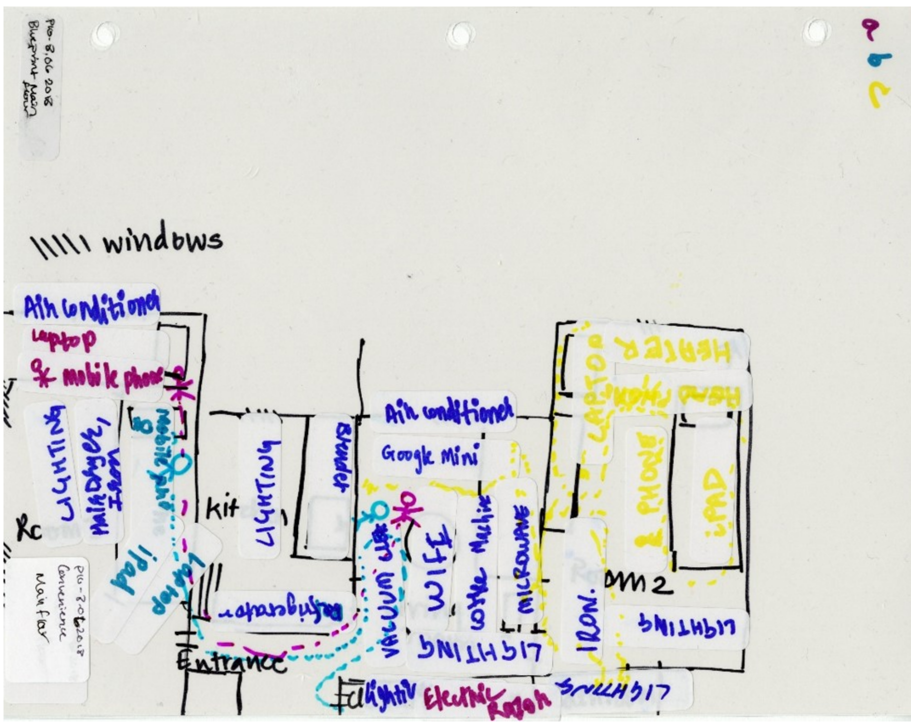


Figure 17: The color blocked annotations of the floor plan shows how family members spatially manage family norms.

B Bibliography

References

[1] *Home Possessions: Material Culture behind Closed Doors*. Berg Publishers, Oxford, UK, 2001.

[2] *Fast-Forward Family: Home, Work and Relationships in Middle-Class America*. University of California Press, United States, 2013.

[3] Iffft helps your apps and devices work together, 2019.

[4] Google nest, build your connected home, 2021.

[5] Smart home products - lighting, wi-fi speakers, blinds - ikea, 2021.

[6] Smart lighting | phillips hue, 2021.

[7] Abdul, A., Vermeulen, J., Wang, D., Lim, B. Y., and Kankanhalli, M. Trends and trajectories for explainable, accountable and intelligible systems: An hci research agenda. In *Proceedings of the 2018 CHI conference on human factors in computing systems* (2018), pp. 1–18.

[8] Abtahi, P., Ding, V., Yang, A. C., Bruzzese, T., Romanos, A. B., Murnane, E. L., Follmer, S., and Landay, J. A. Understanding physical practices and the role of technology in manual self-tracking. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 4*, 4 (2020), 1–24.

[9] Ambe, A., Brereton, M., Soro, A., and Roe, P. Technology individuation: the foibles of augmented everyday objects. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (New York, New York, 2017), ACM, pp. 6632–6644.

[10] Ames, M., Go, J., Kaye, J. J., and Spasojevic, M. Understanding technology choices and values through social class. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work* (New York, NY, 2011), ACM, pp. 55–64.

[11] Annett, M., Grossman, T., Wigdor, D., and Fitzmaurice, G. Moveablemaker: facilitating the design, generation, and assembly of moveable papercraft. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (2015), pp. 565–574.

[12] Arai, T., Aust, D., and Hudson, S. E. Paperlink: a technique for hyperlinking from real paper to electronic content. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (1997), p. 327–334.

[13] Arawjo, I. To write code: The cultural fabrication of programming notation and practice. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020), pp. 1–15.

[14] Arnold, J., Graesch, A., Ragazzini, E., and Ochs, E. *Life at Home in the Twenty-first Century*. Cotsen Institute Press, Los Angeles, CA, 2012.

[15] Arora, N., Zhang, S. L., Shahmiri, F., Osorio, D., Wang, Y.-C., Gupta, M., Wang, Z., Starner, T., Wang, Z. L., and Abowd, G. D. Saturn: A thin and flexible self-powered microphone leveraging triboelectric nanogenerator. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2*, 2 (2018), 60.

[16] Avrahami, D., and Hudson, S. E. Forming interactivity: a tool for rapid prototyping of physical interactive products. In *Proceedings of the 4th conference on Designing interactive systems: processes, practices, methods, and techniques* (2002), pp. 141–146.

[17] Back, M., Cohen, J., Gold, R., Harrison, S., and Minneman, S. Listen reader: an electronically augmented paper-based book. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2001), ACM, pp. 23–29.

[18] Barthel, R., Leder Mackley, K., Hudson-Smith, A., Karpovich, A., De Jode, M., and Speed, C. An internet of old things as an augmented memory system. *Personal and ubiquitous computing 17*, 2 (2013), 321–333.

[19] Biehl, J. T., Girgensohn, A., and Patel, M. Achieving accurate room-level indoor location estimation with emerging iot networks. In *Proceedings of the 9th International Conference on the Internet of Things* (2019), pp. 1–8.

[20] Billinghurst, M., Kato, H., and Poupyrev, I. The magicbook: a transitional ar interface. *Computers & Graphics 25*, 5 (2001), 745–753.

[21] Blackwell, A. F. Psychological issues in end-user programming. In *End user development*. Springer, 2006, pp. 9–30.

[22] Blevis, E. Sustainable interaction design: invention & disposal, renewal & reuse. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2007), pp. 503–512.

[23] Blevis, E., and Stolterman, E. Ensoulment and sustainable interaction design. *Proceedings of IASDR, Hongkong* (2007).

[24] Blumstein, P., and Schwartz, P. *American Couples*. Williams Morrow and Company, Inc., New York, NY, 1983.

[25] Borning, A. H. Classes versus prototypes in object-oriented languages. In *Proceedings of 1986 ACM Fall joint computer conference* (1986), pp. 36–40.

[26] Bowser, A., Novak, A., Parker, A., and Spaulding, J. The potential for low-cost and open source hardware solutions to scale. Woodrow Wilson International Center for Scholars, Washington D.C., USA, 2021.

[27] Brand, S. *How buildings learn: What happens after they're built*. Penguin, 1995.

[28] Braun, V., and Clarke, V. Thematic analysis.

[29] Brich, J., Walch, M., Rietzler, M., Weber, M., and Schaub, F. Exploring end user programming needs in home automation. *ACM Transactions on Computer-Human Interaction (TOCHI) 24*, 2 (2017), 1–35.

[30] Bridgens, B., Powell, M., Farmer, G., Walsh, C., Reed, E., Royapoor, M., Gosling, P., Hall, J., and Heidrich, O. Creative upcycling: Reconnecting people, materials and place through making. *Journal of Cleaner Production 189* (2018), 145–154.

[31] Bronzino, F., Feamster, N., Liu, S., Saxon, J., and Schmitt, P. Mapping the digital divide: before, during, and after covid-19. In *TPRC48: The 48th Research Conference on Communication, Information and Internet Policy* (2021).

[32] Buechley, L., Eisenberg, M., Catchen, J., and Crockett, A. The lilypad arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2008), pp. 423–432.

[33] Buechley, L., Hendrix, S., and Eisenberg, M. Paints, paper, and programs: first steps toward the computational sketchbook. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction* (2009), pp. 9–12.

[34] Buechley, L., and Hill, B. M. Lilypad in the wild: how hardware's long tail is supporting new engineering and design communities. In *Proceedings of the 8th ACM conference on designing interactive systems* (2010), pp. 199–207.

[35] bum Woo, J. User experiences in do-it-yourself-style smart homes. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (New York, NY, 2015), ACM, pp. 779–790.

[36] Bureau, U. C. 2012-2016 american community survey 5-year estimates, 2016.

[37] Bureau, U. C. Characteristics of same-sex couple households, 2016.

[38] Bureau, U. C. Us census bureau quick facts, July 2017.

[39] Bush, V., et al. As we may think. *The Atlantic Monthly 176*, 1 (1945), 101–108.

[40] Buxton, W. Chunking and phrasing and the design of human-computer dialogues. In *Readings in Human–Computer Interaction*. Elsevier, 1995, pp. 494–499.

[41] Calder, P. R., and Linton, M. A. Glyphs: Flyweight objects for user interfaces. In *Proceedings of the 3rd annual ACM SIGGRAPH symposium on User interface software and technology* (1990), pp. 92–101.

[42] Castelli, N., Ogonowski, C., Jakobi, T., Stein, M., Stevens, G., and Wulf, V. What happened in my home? an end-user development approach for smart home data visualization. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (2017), pp. 853–866.

[43] Chang, Z., Ta, T. D., Narumi, K., Kim, H., Okuya, F., Li, D., Kato, K., Qi, J., Miyamoto, Y., Saito, K., et al. Kirigami haptic swatches: Design methods for cut-and-fold haptic feedback mechanisms. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020), pp. 1–12.

[44] Chen, C., Howard, D., Zhang, S. L., Do, Y., Sun, S., Cheng, T., Wang, Z. L., Abowd, G. D., and Oh, H. Spin (self-powered paper interfaces) bridging triboelectric nanogenerator with folding paper creases. In *Proceedings of the Fourteenth International Conference on Tangible, Embedded, and Embodied Interaction* (2020), pp. 431–442.

[45] Chen, X., Coros, S., Mankoff, J., and Hudson, S. E. Encore: 3d printed augmentation of everyday objects with printed-over, affixed and interlocked attachments. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (2015), ACM, pp. 73–82.

[46] Cheon, E., and Su, N. M. Staged for living: Negotiating objects and their values over a porous boundary. *Proceedings of the ACM on Human-Computer Interaction 2*, CSCW (2018), 1–24.

[47] Chetty, M., Banks, R., Harper, R., Regan, T., Sellen, A., Gkantsidis, C., Karagiannis, T., and Key, P. Who's hogging the bandwidth: the consequences of revealing the invisible in the home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2010), pp. 659–668.

[48] Chetty, M., Brush, A. B., Meyers, B. R., and Johns, P. It's not easy being green: understanding home computer power management. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2009), pp. 1033–1042.

[49] Chetty, M., Sung, J.-Y., and Grinter, R. How smart homes learn: The evolution of the networked home and household. In *Proceedings of the International Conference on Ubiquitous Computing* (Berlin, Germany, 2007), Springer, pp. 127–144.

[50] Clark, A., and Dünser, A. An interactive augmented reality coloring book. In *2012 IEEE Symposium on 3D User Interfaces (3DUI)* (2012), IEEE, pp. 7–10.

[51] Crabtree, A., and Rodden, T. Domestic routines and design for the home. *Computer Supported Cooperative Work 13*, 2 (2004), 191–220.

[52] Crouch, I. The horror of amazon's new dash button, 2015.

[53] Cybulski, J. S., Clements, J., and Prakash, M. Foldscope: origami-based paper microscope. *PloS one 9*, 6 (2014), e98781.

[54] Dasgupta, S., Hale, W., Monroy-Hernández, A., and Hill, B. M. Remixing as a pathway to computational thinking. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing* (2016), pp. 1438–1449.

[55] Davidoff, S., Bloomberg, C., Li, I. A. R., Mankoff, J., and Fussell, S. R. The book as user interface: lowering the entry cost to email for elders. In *CHI'05 Extended Abstracts on Human Factors in Computing Systems* (2005), pp. 1331–1334.

[56] Davidoff, S., Zimmerman, J., and Dey, A. Principles of smart home control. In *Proceedings of the 8th International Conference on Ubiquitous Computing* (Berlin, Germany, 2006), Springer, pp. 19–34.

[57] Davidoff, S., Zimmerman, J., and Dey, A. K. How routine learners can support family coordination. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2010), pp. 2461–2470.

[58] De Roeck, D., Slegers, K., Criel, J., Godon, M., Claeys, L., Kilpi, K., and Jacobs, A. I would diyse for it!: a manifesto for do-it-yourself internet-of-things creation. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design* (2012), ACM, pp. 170–179.

[59] Desjardins, A., Key, C., Biggs, H. R., and Aschenbeck, K. Bespoke booklets: A method for situated co-speculation. In *Proceedings of the 2019 on Designing Interactive Systems Conference* (2019), pp. 697–709.

[60] Desjardins, A., Wakkary, R., and Odom, W. Investigating genres and perspectives in hci research on the home. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (2015), pp. 3073–3082.

[61] Dey, A. K. Providing architectural support for building context-aware applications.

[62] Dey, A. K., Sohn, T., Streng, S., and Kodama, J. icap: Interactive prototyping of context-aware applications. In *International conference on pervasive computing* (2006), Springer, pp. 254–271.

[63] Dillahunt, T., Mankoff, J., Paulos, E., and Fussell, S. It's not all about green: energy use in low-income communities. In *Proceedings of the 11th International Conference on Ubiquitous Computing* (New York, NY, 2009), pp. 255–264.

[64] Dillahunt, T. R., Kameswaran, V., Li, L., and Rosenblat, T. Uncovering the values and constraints of real-time ridesharing for low-resource populations. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (2017), pp. 2757–2769.

[65] Dombrowski, L., Harmon, E., and Fox, S. Social justice-oriented interaction design: Outlining key design strategies and commitments. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems* (2016), ACM, pp. 656–671.

[66] Dünser, A., and Hornecker, E. Lessons from an ar book study. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction* (2007), pp. 179–182.

[67] Edwards, K., and Grinter, R. At home with ubiquitous computing: 7 challenges. In *Proceedings of the International Conference on Ubiquitous Computing* (Berlin,Germany, 2001), Springer, pp. 256–272.

[68] Edwards, W. K., Newman, M. W., and Poole, E. S. The infrastructure problem in hci. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2010), ACM, pp. 423–432.

[69] Eisenberg, M., Eisenberg, A., Buechley, L., and Elumeze, N. Invisibility considered harmful: Revisiting traditional principles of ubiquitous computing in the context of education. In *2006 Fourth IEEE International Workshop on Wireless, Mobile and Ubiquitous Technology in Education (WMTE'06)* (2006), IEEE, pp. 103–110.

[70] Elvy, S.-A. Contracting in the age of the internet of things: Article 2 of the ucc and beyond. *Hofstra L. Rev. 44* (2015), 839.

[71] Fedosov, A., Lampinen, A., Odom, W., and Huang, E. M. A dozen stickers on a mailbox: Physical encounters and digital interactions in a local sharing community. *Proceedings of the ACM on Human-Computer Interaction 4*, CSCW3 (2021), 1–23.

[72] Forlizzi, J. How robotic products become social products: An ethnographic study of cleaning in the home. In *Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction* (New York, NY, 2007), ACM, pp. 129–136.

[73] Forlizzi, J. The product ecology: Understanding social product use and supporting design culture. *International Journal of design 2*, 1 (2008).

[74] Freed, N., Qi, J., Setapen, A., Breazeal, C., Buechley, L., and Raffle, H. Sticking together: handcrafting personalized communication interfaces. In *Proceedings of the 10th International Conference on Interaction Design and Children* (2011), pp. 238–241.

[75] Froehlich, J., Findlater, L., and Landay, J. The design of eco-feedback technology. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2010), pp. 1999–2008.

[76] Gantt, M., and Nardi, B. A. Gardeners and gurus: patterns of cooperation among cad users. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (1992), ACM, pp. 107–117.

[77] Gao, C., Li, Y., and Zhang, X. Livetag: Sensing human-object interaction through passive chipless wifi tags. In *USENIX Symposium on Networked Systems Design and Implementation* (Renton, WA, 2018), USENIX Association, pp. 533–546.

[78] Garcia, J., Tsandilas, T., Agon, C., and Mackay, W. Interactive paper substrates to support musical creation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2012), pp. 1825–1828.

[79] Gershenfeld, N., Krikorian, R., and Cohen, D. The internet of things. *Scientific American 291*, 4 (2004), 76–81.

[80] Ghiani, G., Manca, M., Paternò, F., and Santoro, C. Personalization of context-dependent applications through trigger-action rules. *ACM Transactions on Computer-Human Interaction (TOCHI) 24*, 2 (2017), 14.

[81] Gibson, J. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, 1979.

[82] Goldberg, A., and Robson, D. *Smalltalk-80: the language and its implementation*. Addison-Wesley Longman Publishing Co., Inc., 1983.

[83] Goodman, E., and Rosner, D. From garments to gardens: negotiating material relationships online and 'by hand'. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2011), pp. 2257–2266.

[84] Greenberg, H., Gong, M., Magleby, S., and Howell, L. Identifying links between origami and compliant mechanisms. *Mechanical Sciences 2*, 2 (2011), 217–225.

[85] Greenberg, S., and Fitchett, C. Phidgets: easy development of physical interfaces through physical widgets. In *Proceedings of the 14th annual ACM symposium on User interface software and technology* (2001), pp. 209–218.

[86] Grinter, R. E., Edwards, W. K., Newman, M. W., and Ducheneaut, N. The work to make a home network work. In *ECSCW 2005* (2005), Springer, pp. 469–488.

[87] Grudin, J. A moving target: The evolution of hci. *The human-computer interaction handbook: Fundamentals, evolving technologies, and emerging applications* (2008), 1–24.

[88] Guimbretiere, F. Paper augmented digital documents. In *Proceedings of the 16th Annual ACM symposium on User Interface Software and Technology* (2003), pp. 51–60.

[89] Gummeson, J., Mccann, J., Yang, C. J., Ranasinghe, D., Hudson, S., and Sample, A. Rfid light bulb: Enabling ubiquitous deployment of interactive rfid systems. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 1*, 2 (2017), 12.

[90] Hargreaves, T., and Wilson, C. Introduction: smart homes and their users. In *Smart homes and their users*. Springer, 2017, pp. 1–14.

[91] Hargreaves, T., Wilson, C., and Hauxwell-Baldwin, R. Learning to live in a smart home. *Building Research & Information 46*, 1 (2018), 127–139.

[92] Hartmann, B., Klemmer, S. R., Bernstein, M., Abdulla, L., Burr, B., Robinson-Mosher, A., and Gee, J. Reflective physical prototyping through integrated design, test, and analysis. In *Proceedings of the 19th annual ACM symposium on User interface software and technology* (2006), pp. 299–308.

[93] Heiner, J. M., Hudson, S. E., and Tanaka, K. Linking and messaging from real paper in the paper pda. In *Proceedings of the 12th Annual ACM Symposium on User interface software and technology* (1999), ACM, pp. 179–186.

[94] Hendrix, S. L. *Popup workshop: computationally enhanced paper engineering for children*. PhD thesis, University of Colorado, Boulder, 2008.

[95] Hinckley, K., Guimbretiere, F., Agrawala, M., Apitz, G., and Chen, N. Phrasing techniques for multi-stroke selection gestures. In *Graphics Interface* (2006), pp. 147–154.

[96] Hodges, S., and Chen, N. Long tail hardware: Turning device concepts into viable low volume products. *IEEE Computer Architecture Letters 18*, 04 (2019), 51–59.

[97] Hodges, S., Villar, N., Chen, N., Chugh, T., Qi, J., Nowacka, D., and Kawahara, Y. Circuit stickers: peel-and-stick construction of interactive electronic prototypes. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2014), ACM, pp. 1743–1746.

[98] Horn, M. S. The role of cultural forms in tangible interaction design. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction* (2013), pp. 117–124.

[99] Horn, M. S., AlSulaiman, S., and Koh, J. Translating roberto to omar: computational literacy, stickerbooks, and cultural forms. In *Proceedings of the 12th International Conference on Interaction Design and Children* (2013), pp. 120–127.

[100] Hornecker, E., and Buur, J. Getting a grip on tangible interaction: a framework on physical space and social interaction. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (2006), pp. 437–446.

[101] Huang, J., and Cakmak, M. Supporting mental model accuracy in trigger-action programming. In *Proceedings of the 2015 acm international joint conference on pervasive and ubiquitous computing* (2015), pp. 215–225.

[102] Humble, J., Crabtree, A., Hemmings, T., Åkesson, K.-P., Koleva, B., Rodden, T., and Hansson, P. Playing with the bits: User-configuration of ubiquitous domestic environments. In *International Conference on Ubiquitous Computing* (2003), Springer, pp. 256–263.

[103] Iacucci, G., Kuutti, K., and Ranta, M. On the move with a magic thing: role playing in concept design of mobile services and devices. In *Proceedings of the 3rd conference on Designing interactive systems: processes, practices, methods, and techniques* (2000), ACM, pp. 193–202.

[104] Inman, S., and Ribes, D. Beautiful seams: Strategic revelations and concealments. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (2019), pp. 1–14.

[105] Ishii, H., and Ulmer, B. Tangible bits: Towards seamless interfaces between people, bits, and atoms. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, New York, 1997), ACM, pp. 234–241.

[106] Jacobs, J., Brandt, J., Mech, R., and Resnick, M. Extending manual drawing practices with artist-centric programming tools. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), pp. 1–13.

[107] Jacobs, J., and Buechley, L. Codeable objects: computational design and digital fabrication for novice programmers. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2013), pp. 1589–1598.

[108] Jensen, R. H., Strengers, Y., Kjeldskov, J., Nicholls, L., and Skov, M. B. Designing the desirable smart home: A study of household experiences and energy consumption impacts. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), ACM, p. 4.

[109] Johnson, W., Jellinek, H., Klotz Jr, L., Rao, R., and Card, S. K. Bridging the paper and electronic worlds: the paper user interface. In *Proceedings of the INTERACT'93 and CHI'93 Conference on Human Factors in Computing Systems* (1993), pp. 507–512.

[110] Jung, H., Stolterman, E., Ryan, W., Thompson, T., and Siegel, M. Toward a framework for ecologies of artifacts: How are digital artifacts interconnected within a personal life? In *Proceeding of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, 2008), ACM, pp. 201–210.

[111] Karagozler, M. E., Poupyrev, I., Fedder, G. K., and Suzuki, Y. Paper generators: harvesting energy from touching, rubbing and sliding. In *Proceedings of the 26th annual ACM symposium on User interface software and technology* (2013), pp. 23–30.

[112] Kawahara, Y., Hodges, S., Cook, B., Zhang, C., and Abowd, G. Instant inkjet circuits: Lab-based inkjet printing to support rapid prototyping of ubicomp devices. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (New York, New York, 2013), ACM, pp. 363–372.

[113] Kay, A., and Goldberg, A. Personal dynamic media. *Computer 10*, 3 (1977), 31–41.

[114] Kelleher, C., and Pausch, R. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys (CSUR) 37*, 2 (2005), 83–137.

[115] Key, C., Browne, F., Taylor, N., and Rogers, J. Proceed with care: Reimagining home iot through a care perspective. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (2021), pp. 1–15.

[116] Kidd, C. D., Orr, R., Abowd, G. D., Atkeson, C. G., Essa, I. A., MacIntyre, B., Mynatt, E., Starner, T. E., and Newstetter, W. The aware home: A living laboratory for ubiquitous computing research. In *International workshop on cooperative buildings* (1999), Springer, pp. 191–198.

[117] Kim, S., and Paulos, E. Practices in the creative reuse of e-waste. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2011), pp. 2395–2404.

[118] Klemmer, S. R., Graham, J., Wolff, G. J., and Landay, J. A. Books with voices: paper transcripts as a physical interface to oral histories. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2003), pp. 89–96.

[119] Klemmer, S. R., Li, J., Lin, J., and Landay, J. A. Papier-mache: toolkit support for tangible input. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2004), pp. 399–406.

[120] Ko, A. J., Abraham, R., Beckwith, L., Blackwell, A., Burnett, M., Erwig, M., Scaffidi, C., Lawrance, J., Lieberman, H., Myers, B., et al. The state of the art in end-user software engineering. *ACM Computing Surveys (CSUR) 43*, 3 (2011), 1–44.

[121] Kobayashi, H., Kresling, B., and Vincent, J. F. The geometry of unfolding tree leaves. *Proceedings of the Royal Society of London. Series B: Biological Sciences 265*, 1391 (1998), 147–154.

[122] Koshy, V., Park, J. S. S., Cheng, T.-C., and Karahalios, K. "we just use what they give us": Understanding passenger user perspectives in smart homes. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (2021), pp. 1–14.

[123] Kovacs, R., Seufert, A., Wall, L., Chen, H.-T., Meinel, F., Müller, W., You, S., Brehm, M., Striebel, J., Kommana, Y., et al. Trussfab: fabricating sturdy large-scale structures on desktop 3d printers. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (2017), ACM, pp. 2606–2616.

[124] Krasner, G. E., Pope, S. T., et al. A description of the model-view-controller user interface paradigm in the smalltalk-80 system. *Journal of object oriented programming 1*, 3 (1988), 26–49.

[125] Kumar, V., Koehl, L., and Zeng, X. A fully yarn integrated tag for tracking the international textile supply chain. *Journal of Manufacturing Systems 40* (2016), 76–86.

[126] Kurlander, D., Cypher, A., and Halbert, D. C. *Watch what I do: programming by demonstration*. MIT press, 1993.

[127] Lalau-Keraly, C., Daniel, G., Lee, J., and Schwartz, D. Peel-and-stick sensors powered by directed rf energy. In *ASME 2017 International Technical Conference and Exhibition on Packaging and Integration of Electronic and Photonic Microsystems collocated with the ASME 2017 Conference on Information Storage and Processing Systems* (2017), American Society of Mechanical Engineers, pp. V001T03A005–V001T03A005.

[128] Lang, R. J., Magleby, S., and Howell, L. Single degree-of-freedom rigidly foldable cut origami flashers. *Journal of Mechanisms and Robotics 8*, 3 (2016).

[129] Lareau, A. *Unequal Childhoods*. University of California Press, Berkley, CA, 2011.

[130] Lawrence-Lightfoot, S. *Respect: an Exploration*. Perseus Books, New York, NY, 2000.

[131] Lawrence-Lightfoot, S., and Davis, J. H. *The Art and Science of Portraiture*. Jossey-Bass, San Francisco, CA, 1997.

[132] Lazar, A., Edasis, C., and Piper, A. M. Supporting people with dementia in digital social sharing. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (2017), pp. 2149–2162.

[133] Lazar, J., Feng, J. H., and Hochheiser, H. *Research methods in human-computer interaction*. Morgan Kaufmann, 2017.

[134] Lee, B., Han, G., Park, J., and Saakes, D. Consumer to creator: How households buy furniture to inform design and fabrication interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, 2017), ACM.

[135] Lee, J. C., Avrahami, D., Hudson, S. E., Forlizzi, J., Dietz, P. H., and Leigh, D. The calder toolkit: wired and wireless components for rapidly prototyping interactive devices. In *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques* (2004), pp. 167–175.

[136] Lee, J. C., Hudson, S. E., and Tse, E. Foldable interactive displays. In *Proceedings of the 21st annual ACM symposium on user interface software and technology* (2008), pp. 287–290.

[137] Lee, M. L., and Dey, A. K. Sensor-based observations of daily living for aging in place. *Personal and Ubiquitous Computing 19*, 1 (2015), 27–43.

[138] Li, H., Brockmeyer, E., Carter, E., Fromm, J., Hudson, S., Patel, S., and Sample, A. Paperid: A technique for drawing functional battery-free wireless interfaces on paper. In *Proceeding of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, 2016), ACM, pp. 5885–5896.

[139] Li, H., Ye, C., and Sample, A. P. Idsense: A human object interaction detection system based on passive uhf rfid. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (2015), ACM, pp. 2555–2564.

[140] Licklider, J. C. Man-computer symbiosis. *IRE transactions on human factors in electronics*, 1 (1960), 4–11.

[141] Mackay, W. E. *Users and customizable software: A co-adaptive phenomenon*. PhD thesis, Massachusetts Institute of Technology, 1990.

[142] Mackay, W. E. Is paper safer? the role of paper flight strips in air traffic control. *ACM Transactions on Computer-Human Interaction (TOCHI) 6*, 4 (1999), 311–340.

[143] Mackay, W. E., Pothier, G., Letondal, C., Bøegh, K., and Sørensen, H. E. The missing link: augmenting biology laboratory notebooks. In *Proceedings of the 15th annual ACM symposium on User interface software and technology* (2002), pp. 41–50.

[144] MacLean, A., Carter, K., Lövstrand, L., and Moran, T. User-tailorable systems: pressing the issues with buttons. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (1990), pp. 175–182.

[145] Mankoff, J., Hudson, S. E., and Abowd, G. D. Interaction techniques for ambiguity resolution in recognition-based interfaces. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*. 2000, pp. 11–20.

[146] Mankoff, J., Hudson, S. E., and Abowd, G. D. Providing integrated toolkit-level support for ambiguity in recognition-based interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2000), pp. 368–375.

[147] Massey, D., and Denton, N. *American Apartheid: Segregation and the Making of the Underclass*. Harvard University Press, Cambridge, MA, 1993.

[148] McDonough, W., and Braungart, M. *The upcycle: Beyond sustainability–designing for abundance*. Macmillan, 2013.

[149] McGrath, R. E., Craig, A. B., Bock, D., and Rocha, R. Augmented reality for an ethnobotany workbook. Tech. rep., 2011.

[150] Mellis, D. A., Jacoby, S., Buechley, L., Perner-Wilson, H., and Qi, J. Microcontrollers as material: crafting circuits with paper, conductive ink, electronic components, and an" untoolkit". In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction* (2013), pp. 83–90.

[151] Mennicken, S., and Huang, E. M. Hacking the natural habitat: an in-the-wild study of smart homes, their development, and the people who live in them. In *Proceedings of the 10th international conference on Pervasive Computing* (Berlin, Germany, 2012), Springer, pp. 143–160.

[152] Menzel, P. *Material World: a Global Family Portrait*. Sierra Club Books, San Francisco, CA, 1994.

[153] Myers, B., Hudson, S. E., and Pausch, R. Past, present, and future of user interface software tools. *ACM Transactions on Computer-Human Interaction (TOCHI) 7*, 1 (2000), 3–28.

[154] Myers, B. A. A new model for handling input. *ACM Transactions on Information Systems (TOIS) 8*, 3 (1990), 289–320.

[155] Mynatt, E. D., Rowan, J., Craighill, S., and Jacobs, A. Digital family portraits: supporting peace of mind for extended family members. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2001), ACM, pp. 333–340.

[156] Nansen, B., van Ryn, L., Vetere, F., Robertson, T., Brereton, M., and Dourish, P. An internet of social things. In *Proceedings of the 26th Australian Computer-Human Interaction Conference on Designing Futures: the Future of Design* (2014), ACM, pp. 87–96.

[157] Nardi, B. A. *A small matter of programming: perspectives on end user computing*. MIT press, 1993.

[158] Nardi, B. A., O'Day, V., and O'Day, V. L. *Information ecologies: Using technology with heart*. Mit Press, 1999.

[159] Nelson, M. *Parenting Out of Control*. New York University Press, New York, NY, 2010.

[160] Newman, W. M. A system for interactive graphical programming. In *Proceedings of the April 30–May 2, 1968, spring joint computer conference* (1968), pp. 47–54.

[161] Norrie, M. C., Signer, B., and Weibel, N. General framework for the rapid development of interactive paper applications. In *Proc. CoPADD* (2006), vol. 6, pp. 9–12.

[162] Norton, M., Mochon, D., and Ariely, D. The ikea effect: When labor leads to love. *Journal of Consumer Psychology 22* (2012), 453–460.

[163] Odom, W., Pierce, J., Stolterman, E., and Blevis, E. Understanding why we preserve some things and discard others in the context of interaction design. In *Proceeding of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, 2009), ACM, pp. 1053–1062.

[164] of Labor, W. B., and Statistics. Mothers and families, August 2014.

[165] Oh, H., Kim, J., Morales, C., Gross, M., Eisenberg, M., and Hsi, S. Foldmecha: exploratory design and engineering of mechanical papercraft. In *Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction* (2017), pp. 131–139.

[166] Olberding, S., Soto Ortega, S., Hildebrandt, K., and Steimle, J. Foldio: Digital fabrication of interactive and shape-changing objects with foldable printed electronics. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (2015), pp. 223–232.

[167] O'Leary, K., Dong, T., Haines, J. K., Gilbert, M., Churchill, E. F., and Nichols, J. The moving context kit: designing for context shifts in multi-device experiences. In *Proceedings of the 2017 Conference on Designing Interactive Systems* (2017), pp. 309–320.

[168] Olsen Jr, D. R. Mike: The menu interaction kontrol environment. *ACM Transactions on Graphics (TOG) 5*, 4 (1986), 318–344.

[169] Olsen Jr, D. R. A programming language basis for user interface. *ACM SIGCHI Bulletin 20*, SI (1989), 171–176.

[170] Palay, A. J., Hansen, W. J., Kazar, M. L., Sherman, M., Wadlow, M. G., Neuendorffer, T. P., Stern, Z., Bader, M., and Peters, T. The andrew toolkit-an overview. In *USENIX Winter* (1988), pp. 9–22.

[171] Pane, J. F., and Myers, B. A. Tabular and textual methods for selecting objects from a group. In *Proceeding 2000 IEEE International Symposium on Visual Languages* (2000), IEEE, pp. 157–164.

[172] Parikh, T. S., Javid, P., Ghosh, K., and Toyama, K. Mobile phones and paper documents: Evaluating a new approach for capturing microfinance data in rural india. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (2006), pp. 551–560.

[173] Parker, A., Dosemagen, S., Molloy, J., Bowser, A., and Novak, A. Open hardware: An opportunity to build better science. Woodrow Wilson International Center for Scholars, Washington D.C., USA, 2021.

[174] Pierce, J., and Paulos, E. Second-hand interactions: investigating reacquisition and dispossession practices around domestic objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2011), pp. 2385–2394.

[175] Piper, A. M., Weibel, N., and Hollan, J. D. Tap & play: an end-user toolkit for authoring interactive pen and paper language activities. In *CHI* (2012), vol. 12, p. 149.

[176] Plaisant, C., Clamage, A., Hutchinson, H. B., Bederson, B. B., and Druin, A. Shared family calendars: Promoting symmetry and accessibility. *ACM Transactions on Computer-Human Interaction (TOCHI) 13*, 3 (2006), 313–346.

[177] Preist, C., Schien, D., and Blevis, E. Understanding and mitigating the effects of device and cloud service design decisions on the environmental footprint of digital infrastructure. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (2016), pp. 1324–1337.

[178] Qamar, I. P., Groh, R., Holman, D., and Roudaut, A. Hci meets material science: A literature review of morphing materials for the design of shape-changing interfaces. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), pp. 1–23.

[179] Qi, J., and Buechley, L. Electronic popables: exploring paper-based computing through an interactive pop-up book. In *Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction* (2010), pp. 121–128.

[180] Qi, J., and Buechley, L. Sketching in circuits: designing and building electronics on paper. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2014), pp. 1713–1722.

[181] Qi, J., Demir, A., and Paradiso, J. A. Code collage: Tangible programming on paper with circuit stickers. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (2017), pp. 1970–1977.

[182] Qi, J., Huang, A. b., and Paradiso, J. Crafting technology with circuit stickers. In *Proceedings of the 14th International Conference on Interaction Design and Children* (2015), pp. 438–441.

[183] Rathje, W., and Murphy, C. *Rubbish! The Archaeology of Garbage*. Harper Collins, New York, NY, 1992.

[184] Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., et al. Scratch: programming for all. *Communications of the ACM 52*, 11 (2009), 60–67.

[185] Resnick, M., Myers, B., Nakakoji, K., Shneiderman, B., Pausch, R., Selker, T., and Eisenberg, M. Design principles for tools to support creative thinking.(2005). Tech. rep., 2005.

[186] Resnick, M., and Silverman, B. Some reflections on designing construction kits for kids. In *Proceedings of the 2005 conference on Interaction design and children* (2005), pp. 117–122.

[187] Richardson, H. J. A 'smart house' is not a home: The domestication of icts. *Information Systems Frontiers 11*, 5 (2009), 599–608.

[188] Rodden, T., and Benford, S. The evolution of buildings and implications for the design of ubiquitous domestic environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2003), pp. 9–16.

[189] Rode, J. A., Toye, E. F., and Blackwell, A. F. The fuzzy felt ethnography—understanding the programming patterns of domestic appliances. *Personal and Ubiquitous Computing 8*, 3-4 (2004), 161–176.

[190] Rogers, Y. Moving on from weiser's vision of calm computing: Engaging ubicomp experiences. In *International conference on Ubiquitous computing* (2006), Springer, pp. 404–421.

[191] Rohs, M. Visual code widgets for marker-based interaction. In *25th IEEE International Conference on Distributed Computing Systems Workshops* (2005), IEEE, pp. 506–513.

[192] Rowan, J., and Mynatt, E. D. Digital family portrait field trial: Support for aging in place. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2005), ACM, pp. 521–530.

[193] Salber, D., Dey, A. K., and Abowd, G. D. The context toolkit: Aiding the development of context-enabled applications. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (1999), pp. 434–441.

[194] Schiffer, M., Downing, T., and McCarthy, M. *Waste Not, Want Not: An Ethnoarchaeological Study of Reuse in Tucson, Arizona*. Academic Press, New York, NY, 1981.

[195] Schiffer, M. B. *Studying technological change: A behavioral approach*. University of Utah Press, 2011.

[196] Schilit, B., Adams, N., and Want, R. Context-aware computing applications. In *1994 First Workshop on Mobile Computing Systems and Applications* (1994), IEEE, pp. 85–90.

[197] Schlesinger, A., Edwards, W. K., and Grinter, R. E. Intersectional hci: Engaging identity through gender, race, and class. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (2017), ACM, pp. 5412–5427.

[198] Schmidt, A. Implicit human computer interaction through context. *Personal technologies 4*, 2 (2000), 191–199.

[199] Schwarz, J. *Monte Carlo Methods for Managing Uncertain User Interfaces*. PhD thesis, Carnegie Mellon University, 2014.

[200] Sharova, A. S., Melloni, F., Lanzani, G., Bettinger, C. J., and Caironi, M. Edible electronics: The vision and the challenge. *Advanced Materials Technologies 6*, 2 (2021), 2000757.

[201] Shaw, J. C. Joss: A designer's view of an experimental on-line computing system. In *Proceedings of the October 27-29, 1964, fall joint computer conference, part I* (1964), pp. 455–464.

[202] Shewbridge, R., Hurst, A., and Kane, S. K. Everyday making: identifying future uses for 3d printing in the home. In *Proceedings of the 2014 conference on Designing interactive systems* (2014), ACM, pp. 815–824.

[203] Sloman, A. Interactions between philosophy and artificial intelligence: The role of intuition and non-logical reasoning in intelligence. *Artificial intelligence 2*, 3-4 (1971), 209–225.

[204] Soro, A., Brereton, M., and Roe, P. *Social Internet of Things*. Springer, 2018.

[205] Speilberg, A., Sample, A., Hudson, S., Mankoff, J., and McCann, J. Rapid: A framework for fabricating low-latency interactive objects with rfid tags. In *Proceeding of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, 2016), ACM, pp. 5897–5908.

[206] Stifelman, L., Arons, B., and Schmandt, C. The audio notebook: paper and pen interaction with structured speech. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2001), pp. 182–189.

[207] Stifelman, L. J. Augmenting real-world objects: A paper-based audio notebook. In *Conference companion on Human factors in computing systems* (1996), pp. 199–200.

[208] Suh, H., Shahriaree, N., Heckler, E., and Kientz, J. Developing and validating the user burden scale: a tool for assessing user burden in computing systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing* (New York, NY, 2016), ACM, pp. 3988–3999.

[209] Sung, K., Cooper, T., and Kettley, S. An exploratory study on the links between individual upcycling, product attachment, and product longevity. Nottingham Trent University.

[210] Sutherland, I. E. Sketchpad a man-machine graphical communication system. *Simulation 2*, 5 (1964), R–3.

[211] Swan, L., Taylor, A. S., and Harper, R. Making place for clutter and other ideas of home. *ACM Transactions on Computer-Human Interaction (TOCHI) 15*, 2 (2008), 1–24.

[212] Sylla, C., Branco, P., Gonçalves, S., Coutinho, C., and Brito, P. T-books: merging traditional storybooks with electronics. In *Proceedings of the 11th International Conference on Interaction Design and Children* (2012), pp. 323–326.

[213] Taylor, A. S., Harper, R., Swan, L., Izadi, S., Sellen, A., and Perry, M. Homes that make us smart. *Personal and Ubiquitous Computing 11*, 5 (2007), 383–393.

[214] Taylor, A. S., Wyche, S. P., and Kaye, J. Pottering by design. In *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges* (2008), pp. 363–372.

[215] Tholander, J., and Normark, M. Crafting personal information-resistance, imperfection, and self-creation in bullet journaling. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020), pp. 1–13.

[216] Truong, K. N., Huang, E. M., and Abowd, G. D. Camp: A magnetic poetry interface for end-user programming of capture applications for the home. In *International Conference on Ubiquitous Computing* (2004), Springer, pp. 143–160.

[217] Uncapher, K. W. Oral history interview with keith w. uncapher.

[218] Ur, B., McManus, E., Pak Yong Ho, M., and Littman, M. L. Practical trigger-action programming in the smart home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2014), pp. 803–812.

[219] Ur, B., Pak Yong Ho, M., Brawner, S., Lee, J., Mennicken, S., Picard, N., Schulze, D., and Littman, M. L. Trigger-action programming in the wild: An analysis of 200,000 ifttt recipes. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (2016), ACM, pp. 3227–3231.

[220] Vogel, S. *Cats' paws and catapults: Mechanical worlds of nature and people*. WW Norton & Company, 2000.

[221] Vyas, D. Domestic artefacts: Sustainability in the context of indian middle class. In *Proceedings of the 4th International Conference on Intercultural Collaboration* (New York, NY, 2012), ACM, pp. 119–128.

[222] Wang, G., Cheng, T., Do, Y., Yang, H., Tao, Y., Gu, J., An, B., and Yao, L. Printed paper actuator: A low-cost reversible actuation and sensing method for shape changing interfaces. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), pp. 1–12.

[223] Watanabe, J.-i., Mochizuki, A., and Horry, Y. Bookisheet: bendable device for browsing content using the metaphor of leafing through the pages. In *Proceedings of the 10th international conference on Ubiquitous computing* (2008), pp. 360–369.

[224] Watson, C., Kirkham, R., and Kharrufa, A. Pip kit: An exploratory investigation into using lifelogging to support disability benefit claimants. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020), pp. 1–14.

[225] Weiser, M. The computer of the 21st century. *Mobile Computing and Communications Review 3*, 3 (1991).

[226] Wellner, P. The digitaldesk calculator: tangible manipulation on a desk top display. In *Proceedings of the 4th annual ACM symposium on User interface software and technology* (1991), pp. 27–33.

[227] Wellner, P. Interacting with paper on the digitaldesk. *Communications of the ACM 36*, 7 (1993), 87–96.

[228] Wessely, M., Tsandilas, T., and Mackay, W. E. Shape-aware material: Interactive fabrication with shapeme. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (2018), pp. 127–139.

[229] Whitesides, G. The frugal way. *The Economist 17* (2011).

[230] Wie, D. S., Zhang, Y., Kim, M. K., Kim, B., Park, S., Kim, Y.-J., Irazoqui, P. P., Zheng, X., Xu, B., and Lee, C. H. Wafer-recyclable, environment-friendly transfer printing for large-scale thin-film nanoelectronics. In *Proceedings of the National Academy of Science* (2018), vol. 115, PNAS.

[231] Wilkins, D. J., Chitchyan, R., and Levine, M. Peer-to-peer energy markets: Understanding the values of collective and community trading. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020), pp. 1–14.

[232] Wilson, C., and Hargreaves, T. Smart homes and their users: a systematic analysis and key challenges. *Personal and Ubiquitous Computing 19*, 2 (2015).

[233] Winder, B. G., Magleby, S. P., and Howell, L. L. Kinematic representations of pop-up paper mechanisms. *Journal of mechanisms and robotics 1*, 2 (2009).

[234] Yardi, S., and Bruckman, A. Income, race, and class: Exploring socioeconomic differences in family technology use. In *Proceedings of the SIGCHI Conference on Human Factors in Computing* (New York, NY, 2011), ACM, pp. 3041–3050.

[235] Yeh, R. B., Paepcke, A., and Klemmer, S. R. Iterative design and evaluation of an event architecture for pen-and-paper interfaces. In *Proceedings of the 21st annual ACM symposium on User interface software and technology* (2008), pp. 111–120.

[236] Zegura, E., Nazareno, L., and Liu, C. Y. Changes in mobile broadband infrastructure in rural georgia during the covid-19 pandemic. *Available at SSRN 3898212* (2021).

[237] Zhang, Y., Iravantchi, Y., Jin, H., Kumar, S., and Harrison, C. Sozu: Self-powered radio tags for building-scale activity sensing. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (2019), pp. 973–985.

[238] Zhao, Y., Qin, Y., Liu, Y., Liu, S., and Shi, Y. Qook: A new physical-virtual coupling experience for active reading. In *Proceedings of the adjunct publication of the 26th annual ACM symposium on User interface software and technology* (2013), pp. 5–6.

[239] Zheng, C., Gyory, P., and Do, E. Y.-L. Tangible interfaces with printed paper markers. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference* (2020), pp. 909–923.

[240] Zheng, C., Oh, H., Devendorf, L., and Do, E. Y.-L. Sensing kirigami. In *Proceedings of the 2019 on Designing Interactive Systems Conference* (2019), pp. 921–934.

[241] Zhu, K., and Zhao, S. Autogami: a low-cost rapid prototyping toolkit for automated movable paper craft. In *Proceedings of the SIGCHI conference on human factors in computing systems* (2013), pp. 661–670.