Designing the Fog: Towards an Intranet of Things

Mathias Funk
Industrial Design Department
Eindhoven University of Technology
Eindhoven, the Netherlands
m.funk@tue.nl

Abstract

The Fog of Things, ubiquitous computing in local contexts, is a reality now. The Internet of Things has arrived, although users use and perceive it rather as Internet of Thing [sic!]. Data and information flows vertically, not horizontally through the connected Everyday and promises of convenience and quality of life are at the mercy of the viability of business models and the benevolence of multi-national commercial entities. This position paper poses that *things* and connectedness can also be re-thought; products and services can be designed differently: bottom-up and with stronger ideals in place. Systems of connected things can be understood as horizontal autonomous networks of nodes, Intranets, that do not or only seldom connect to external entities for the exchange of data. This paper explains how to conceptualize these systems, proposes the use of system properties to address new design challenges, and concludes with an outlook on future work.

Author Keywords

Smart things; Internet of Things; Systems Design; Interaction Design.

ACM Classification Keywords

H.5.m. Information interfaces and presentation.

Introduction

Increasingly computing extends to the fringes of the Internet, towards local contexts that are in close proximity to users and that allow for direct interaction and consumption of ubiquitous services offered by the computing infrastructure. Devices emerged that package new functionality based on connectedness, on information streams and frequent data exchange [1]: tracking, monitoring and measuring, notifying or alerting, adjusting and connecting, to name a few. This has not only implications for new devices and services that enter our personal spaces [2], there are implications for how we see ourselves reflected in such data, often freely accessible and exploitable, now or in the future [3].

Consumer IoT products may disguise as traditional interactive products, but they are inherently more than that: they are connected. And this has serious implications [4]. They require not only more expertise and skills about networked technology to setup and install, configuration of connected products is a disguised maintenance process (similar to what is known from industrial or professional contexts). For example, in the context of a smart home with tens of connected devices, changing the Wifi password becomes a task that might take days instead of minutes, requiring the users to track back obscure configuration methods and parameters – often *per* device. There is the question of compatibility that is beyond sockets, cables and plugs; compatibility no longer guaranteed by international industry-wide standards, compatibility becomes a challenge at the higher level of APIs and connection protocols. These are certainly out of control of users, and often change over time, which requires maintenance, workarounds for older devices, or even new investments. Finally, an installation of connected devices grows over time [5], due to changing needs, maintenance, or replacements [6]. Sometimes a new device is added because it promised new functions that might or might not be complementary to what already exists in the context. This growth adds to the complexity, compatibility issues, functionality overlaps, data that cannot be easily exported and imported again, thereby complicating the operation of a smart home. Especially this last point shows that consumer IoT can become in principle more complex than industrial IoT installations, which are carefully planned, designed and built to clearly specified needs. The emergence of connected devices has clearly raised concerns about privacy and (data) ownership with users [7, 8]. Furthermore, devices are often implemented without proper security mechanism when, for instance, accessing a local wireless access point or communicating data to a cloud appliance. When operating in a technologically uncertain or diverse environment such as homes, offices, and public buildings, commercial IoT devices and systems embody a trade-off between ease of deployment (not even ease of use) and data security. Usually, compromises on deployment would lead to higher costs, so data security is likely to be neglected. Finally, IoT devices might easily live longer than their manufacturers. Consumers expect a 5 to 10-year life, while the manufacturer might be a small company, seed funding initiative or even side-project, and might not exist after one or two years. By now, we have seen large corporations shedding IoT products from their portfolios the moment they become misaligned with grand strategy [9]. From that point onwards, IoT devices that rely on external services might become useless, or at least will not get updates anymore that would ensure compatibility and appropriate functioning and prevent hacking or misuse by adversaries [10].

The abovementioned challenges are well-known for year and attractive for design. However, in the context of design, systems are often misunderstood and misrepresented [11, 12]. Furthermore, end-to-end systems design is often not practiced. Instead, the role of design is limited to interface-related usability and user experience problems, and pure form-giving. When investigating available technology in the consumer space, a clear anti-pattern can be observed: The *Internet of Thing*. These are small-scale systems, often thing-app tandems, which occupy the personal as a singular product extending an invisible, intangible link to the cloud. Examples are everywhere to be found, Internet of Thing devices that only work when they are connected to servers in the cloud, devices that will not work without centrally managed licenses and keys, and control structures that are highly optimized for performance, cost and control, but not for robustness and participative data ownership. Even just a tight coupling between a physical "smart object" and a proximal smart phone app extends the object's capabilities, but with the effect of centralizing control further towards the smart phone and encasing "smart" functionality in a silo of specifically this object-app tandem. For systems design, these are clearly anti-patterns of well-designed systems.

In the remainder of this position paper, we will explain the main challenges for systems design in the context of IoT, and focus on emergent properties in systems of things. The position paper continues with a discussion and concludes with a summary and a section on future work.

From Internet of Thing to Intranet of Things

It is important to understand that IoT technology for consumers in personal spaces is currently still in its infancy and will exhibit flaws that become apparent only over time when technology, business models and services evolve, when the true needs and expectations of consumers become clear. Such a context with the premise of connectedness and ubiquitous computing embedded into the environment is ideal for design: yet, instead of technological frameworks for systems, we need design frameworks for *systems*. In this paper, we propose the *design of connected things* that will *work independently of external services*, that will continue to *serve without an Internet connection*, and that will *not share collected data with external parties*.

Design space

If we want to predict a second or third wave of connected products that better behave, how can this design space be characterized? From a technological angle, what we describe here is an intranet. Devices in an Intranet of Things form a constellation and might use the Internet as an *optional in-bound* source of data, but should not rely on it. Instead of extending their scope towards external services, they extend horizontally: from room to floor to household, from street to neighborhood to community different scopes are imaginable. Apart from this spatial characterization, the scope encompasses an oversee-able collective of actors, people and things.

If we think of designing connected things that form a local constellation, the complexity of conceptualizing, designing, deploying and using becomes larger – especially if not all devices "play together" nicely. If we want to design for a future when devices will primarily exist in (local) constellations, we need to change (1) how designers *look* at products and users or stakeholders in the Intranet of Things, and (2) the way designers understand and *leverage technology* fully.

For (1), future connected products in a constellation of connected things, are intended to be more than just the sum of each individual thing. This is currently not the case. The reason for connectivity is that products are extensions of a cloud-based business processes, not autonomous actors on their users' behalf. Our design vision is that products are created as independent actors that can act in the local context based on shared information and semantics that are exchanged at a local level. They act with autonomy that is a well-designed trait, not a fallback mechanism, nor after-thought. The consequence is that design needs to extend beyond the scope of a single product and take ecologies [13] into account.

For (2) and the foreseeable future, designers need to bridge an important gap: on the one side, there is IoT technology built and deployed in the context that follows the prevalent cloud connectivity paradigm, and on the other side, a new focus on locality and data ownership needs to be nurtured through new technology which promotes different use cases and thus different designs. To bridge this gap, a better, more practical definition of systems of connected things is needed for approach better technological framings.

In the following, we will first elaborate the "Intranet of Things" as a design vision, and then take a more technological perspective towards thing ecologies as designed and evolved constellations of things.

Things

We assume that all things in a constellation are, to some degree, capable of computing data and using the result in action and communication. Some things might focus on the sensing and transmission of sensor data. Other components fulfill different roles, which require them to process data, filter events, store information, manipulate weights of a learning algorithm and trigger actions. While these capabilities require different configurations in computing power and storage, energy and connectivity might be impacted as well. In general, in the scope of this position paper, we assume that sufficient computation is available at all layers of a system. When a design moves towards production, such requirements can be trimmed to better fit production or business constraints.

This structure is, however, not enough; like cells of an organism, without exchange of fluids, decay sets in. Data and information flows are necessary to turn a static set of components into a live system. The components are the essence from which systems emerge. They determine the nature of the system, it's "product-ness", and specific tool or service qualities. The question what we are designing needs to start with components, rather than the big picture, and then build connections, layers, interconnections, and cross-sections towards the emergence of system properties.

Systems of Connected Things

As described above, a challenge of systems design is that systems are often mischaracterized and thus oversimplified already in early phases of the design process. Simplifications target common convergence points: single products that are functionally overloaded, imbalanced product-app tandems and cloud services with product as crippled physical manifestations. To counter this, product design needs to embrace the concept of “emergent properties”, i.e., desirable properties of a system that a designer might find applicable in a design project. In the following, we describe a non-exhaustive selection of such emergent properties (see Table 1). These properties demonstrate balance between describing desirable traits of a system and relating to possible approaches in systems design to realize these traits. The properties in Table 1 are connected and non-contradictory, i.e., they are formulated from a user point of view and some of them can only be realized in connection to other properties being apparent in the design. The properties are divided into three categories, *structure*, *behavior*, and *interfaces*. For each category, two or three emergent properties are presented and briefly discussed in the context of the category.

	Property	**Application / Example**
Structure	Growth	Changing structure (extension, pruning, re-organization)
	Shared functionality	Functional grouping
Behavior	Emergence through homogeneity	Complex behavior: synchronization, self-organization
	Emergence through heterogeneity	Social behavior (amongst things)
	Cooperation and Negotiation	Dynamic roles, task switching
	Autonomy	Self-contained functionality, independent operation
	Distributed cognition	Shared sense-making, exploration of context, new situations
Interfaces	Adaptation through local learning	Contextualization as adaptation to context
		Personalization as adaptation to user(s)
	(Inter-)locality and Representation	Sharing information (amongst things)
		Sharing information (towards user(s))
	Intentionality	Social behavior (towards user(s))

Table 1: Overview of System Properties categorized into structure, behavior, and interfaces.

Structure

When a system as a constellation of things *grows,* this means that its structure–often called (physical) graph–changes: either the set of nodes is changed by adding, removing or replacing a node, or the linking between nodes changes. Such mechanisms of growth might result from a user purchasing a new device, a faulty device dropping out, or a firmware update. Growth in how things connect is more intricate: things can connect based on their spatial location and proximity, they can form functional groups and through such groups facilitate other emergent behavior such as shared functionality, emergence and redundancy. When things connect, they can also establish hierarchies and functional groups to address external requests, or to balance a heavy load.

Behavior

We know emergence from highly homogeneous systems in nature, e.g., fireflies synchronize their blinking and swarms of birds form and scare away predators that are larger than individual birds in the swarm. Such emergence can be leveraged in design to synchronize concurrent processes in distributed products. At the same time, product can engage in social interaction based on their heterogeneity. Patterns of cooperation and negotiation can help further self-organization, decision making and recovery from fast growth. These behavioral patterns can be complemented by highly autonomous behavior that helps a thing maintain its function during a period of limited connectivity to its peers. This means that the functionality of the system is not only distributed across several components, it means that this functionality is performed regardless of central coordination in a redundant, parallel and autonomous way. Such a design can mitigate the failure of one component which can be balanced out by another; components can "team up" to autonomously recognize failure of one team member and act upon it, and functionality can be provided in different locations simultaneously without coordination. Finally, some systems might exhibit forms of distributed cognition when utilizing different complementary capabilities of things to make sense of a new situation or context.

Interfaces

When things interface with each other, or the context, they can be designed to sense and react to "friction" in interfacing. If a thing encounters friction its environment it can decide to use local adaptation of a particular aspect that the thing embodies. This adaptation basically reduces friction in interoperation between the thing and its counterparts. Different from distributed cognition, this adaptation is a form of local learning that each thing performs on its own. In systems of connected things, the creation, presence, decay, and absence of data determines the system's functionality and structural composition. Data can be raw measurements, control data or commands, or high-level information that influences the experience and decision making of users. As a system property, data refers to sharing of data between things, or the sharing of information between the system and its users. While sharing data with components of a system obviously requires formal interfaces and clear protocols, sharing data with human users through data representations such as visualizations, decorations, physicalizations or even sonifications, requires a similarly careful approach. When looking at interfaces and exchanges of information, the focus shifts towards the data that is gathered, processed and stored in the system, where data flows and whether data flows can be limited to a local, yet systemic scope without leaking out unintentionally. This property, as a consequence, implies that designers need to consider how locality can be achieved without a central and global controller instance. To counter the need to involve global or central control structured, designers can take a hybrid approach and consider inter-locality, i.e., a group of "local" components that cooperatively share data or, more appropriately, models derived from data, that abstract from local details whenever benefits of sharing such information between systems or subsystems emerge. An example is a security system that shares key signifiers of trusted persons amongst components without a central instance storing privacy-sensitive information.

Discussion

The three categories of system properties as explained above pose interesting opportunities for the design of connected things. Utilizing emergent or systemic properties of connected things is a function of (hyper-)

connecting things, augmenting them with computation and redesigning shared functionality with clear “situatedness” and locality in mind. Locality refers to a strong dependence and product relation to the local context – in functionality and operational semantics. Traditional designed products have naturally been local and self-contained (due to a lack of inherent connectivity, computation and data streams). In contrast, most connected interactive products are designed with a central controller, database or service in place without which the product cannot or not operate as well. An emphasis on locality has benefits in better control over privacy-sensitive data, protection against censorship, robustness against connectivity infrastructure problems, and the failure of single components. Furthermore, we can adhere to share information instead of data, and quality instead of bulk. Whenever new data is sensed and collected by components of the system, they should make sense of this data, enrich it with contextual “knowledge” and then reluctantly share it with other system components on their request. This rule is central and applies to components or layers of a system, but also the system as a hierarchical sub-system. Data trickles bottom-up, in gradually more processed and richer forms. However, even though we might be *aware* of possible properties and their opportunities, leveraging them in a design project is hard because technology is missing or existing technologies have not been framing appropriately. Another problem might be that control of decentralized functionality by a user might be more difficult or can only be done indirectly, maintaining a consistent state or, and synchronization across different locations requires again a connection. Processes that have established operating procedures such as updating the system or shutting down operation temporarily can be cumbersome and resource use might be higher through duplication of services and resources locally. Finally, critical mass for emergent behavior is not always given. It is possible to imagine a layer for individual people’s room in a shared apartment, a layer for the apartment or house, and then a layer for the apartment building or neighborhood – each with a different purpose and functionality. But if there are too few things participating, no ecology of things will develop.

Conclusion

In this position paper, we have introduced and discussed a new frontier for designing IoT – the Intranet of Things. While many problems and challenges of IoT systems and products are well-known, this concept is proposed as a new framing that is opposed to the conventional IoT vision that puts cloud control, data leaking and privacy breaches first. T

he Intranet of Things favors locality, decentralized behaviors and creative use of communication pathways between things in a local context. Such communication is meant to share data and to engage in reinforcing behavior and thing-to-thing feedback loops. The creation of such new loops is an interesting field for design, creating awareness and giving access to leverage points that has not existed before.

Future work

To support designers in creative applications of connected things, new technologies and technological framings of existing technologies are required. Building on such grounds, we need to develop frameworks for creating systems prototypes using established design software and hardware platforms that allow for fluent use and rapid iterations through a new systems design process. We hope that through exploration like this position paper and extensive discussions, a new generation of connected things will emerge that comply to our intuitive understanding of things as meaningful parts of the future Everyday.

References

1. Odom, William, Tom Jenkins, Kristina Andersen, Bill Gaver, James Pierce, Anna Vallgårda, Andy Boucher, David Chatting, Janne van Kollenburg, and Kevin Lefeuvre. 2017. Crafting a Place for Attending to the Things of Design at CHI. *interactions* 25. New York, NY, USA: ACM: 52–57. doi:10.1145/3161605.
2. Svanæs, Dag. 2001. Context-Aware Technology: A Phenomenological Perspective. *Human–Computer Interaction* 16. Taylor & Francis: 379–400. doi:10.1207/S15327051HCI16234_17.
3. Zhang, Ben, Nitesh Mor, John Kolb, Douglas S Chan, Ken Lutz, Eric Allman, John Wawrzynek, Edward A Lee, and John Kubiatowicz. 2015. The Cloud is Not Enough: Saving IoT from the Cloud. *7th USENIX Workshop on Hot Topics in Cloud Computing, HotCloud '15, Santa Clara, CA, USA, July 6-7, 2015.*
4. Rogers, Yvonne. 2006. Moving on from Weiser's Vision of Calm Computing: Engaging Ubicomp Experiences. In *Proceedings of the 8th International Conference on Ubiquitous Computing*, 404–421. UbiComp'06. Berlin, Heidelberg: Springer-Verlag. doi:10.1007/11853565_24.
5. Crabtree, Andy, and Peter Tolmie. 2016. A Day in the Life of Things in the Home. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, 1738–1750. CSCW '16. New York, NY, USA: ACM. doi:10.1145/2818048.2819954.
6. Odom, William, James Pierce, Erik Stolterman, and Eli Blevis. 2009. Understanding Why We Preserve Some Things and Discard Others in the Context of Interaction Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1053–1062. CHI '09. New York, NY, USA: ACM. doi:10.1145/1518701.1518862.
7. Churchill, Elizabeth F. 2016. Designing Data Practices. *interactions* 23. New York, NY, USA: ACM: 20–21. doi:10.1145/2983401.
8. Williams, Meredydd, Jason R C Nurse, and Sadie Creese. The Perfect Storm : The Privacy Paradox and the Internet - of - Things: 1–9. doi:10.1109/ARES.2016.25.
9. Perlow, Jason. 2016. All your IoT devices are doomed. *ZDNet*.
10. Abowd, Gregory D. 2012. What Next, Ubicomp?: Celebrating an Intellectual Disappearing Act. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, 31–40. UbiComp '12. New York, NY, USA: ACM. doi:10.1145/2370216.2370222.
11. Ryan, William, Erik Stolterman, Heekyoung Jung, Martin Siegel, Tonya Thompson, and William R Hazlewood. 2009. Device Ecology Mapper: A Tool for Studying Users' Ecosystems of Interactive Artifacts. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, 4327–4332. CHI EA '09. New York, NY, USA: ACM. doi:10.1145/1520340.1520661.
12. Ryan, Alex. 2014. A Framework for Systemic Design. *FORMakademisk* 7. doi:10.7577/formakademisk.787.
13. Jung, Heekyoung, Erik Stolterman, Will Ryan, Tonya Thompson, and Marty Siegel. 2008. Toward a Framework for Ecologies of Artifacts: How Are Digital Artifacts Interconnected Within a Personal Life? In *Proceedings of the 5th Nordic Conference on Human-computer Interaction: Building Bridges*, 201–210. NordiCHI '08. New York, NY, USA: ACM. doi:10.1145/1463160.1463182.