An Animal Care Smart Home System Built on OpenThread

Jingwei Dai
Department of Computer Science
Earlham College
Richmond, Indiana, USA
adai13@earlham.edu

Abstract

The marine aquarium industry has in recent years experimented with IoT (Internet of Things) technologies to automate maintenance and upkeep. However, the most prevalent solutions predominantly rely on Wi-Fi or Bluetooth, both of which come with a range of limitations. These solutions also often operate as isolated ecosystems, posing challenges for seamless integration into established Smart Home platforms like Apple's HomeKit.

This project will explore the latest innovation in IoT connectivity standards, Thread, and its open source implementation, OpenThread. Our aim is to provide an API built on top of OpenThread that empower third-party IoT manufacturers to tailor their Thread-capable products for specific applications within large scale artificial environments, such as marine aquariums and aviaries. In addition, we will develop a smart lighting prototype to demonstrate the distinctive advantages of Thread over the alternatives. Overall, the project will highlight how purpose-built APIs can steer the development of more robust smart animal-care platforms.

Keywords

IoT, HomeKit, Thread, OpenThread, Reef, Marine Aquariums, Smart Lighting

1 Introduction

The current smart home landscape is dominated by three major platforms: Amazon Alexa, Apple HomeKit, and Google Home, all backed by their respective smart assistants. These define the quintessential frontend user experience, because they are what users directly interact with to control their smart homes. Throughout the 2010s, choosing which platform to stick with was an important decision. The platforms were not interoperable: Alexa, Siri, and Google Assistant spoke totally different languages under the hood, and most smart home products could only be connected to one platform at a time. Over the last few years, tech giants have come together to remedy the situation [1]. The result is Matter, an open source interoperability standard that vows to unite the world of IoT [5]. All three platforms have signed on to fully integrate the Matter application layer in the next few years, which means they will eventually sport the same backend with the only difference being UI/UX. When the transition completes, smart home devices conforming to Matter will work under all platforms without extra tinkering. The engineering product of this smart home project will be based on the Apple HomeKit platform, which is already 100% compatible with Matter 1.0 [9].

The platforms, including those that have transitioned to Matter, support multiple connection protocols through which devices talk to the platform backend and each other. As of 2023, there are half a dozen popular protocols in the IoT industry [17]. Major IoT manufacturers Philips and Aqara use ZigBee as the protocol for their entire line of smart home devices [6]. ZigBee devices do not talk to the platforms directly. They can only function when a hub is connected to the internet router to act as a relay for communication between themselves and the platform backend, but hubs from some manufacturers are not compatible with competitors' devices. For example, Aqara's ZigBee implementation veers off the standard by an enough amount that third party devices cannot connect to Aqara's hubs.

However, standardisation through Matter may solve this fragmentation of technology in the IoT industry. Matter recommends Thread, a relatively new protocol that consolidates the advantages of the most popular connection protocols. With the inclusion of a Thread radio in the iPhone 15 [10], we are seeing major effort from industry leaders to promote and facilitate the adoption of this technology.

2 Project Goals

The goal of the Smart Home project is to create an iOS app that manages third party smart home devices in the pet care space. Supported devices are built on OpenThread and can be controlled through either the app or Apple HomeKit. APIs will be provided to guide the development of compatible firmware. The app itself interfaces with HomeKit, exposing essential functions of the devices to Siri and the Home app through predefined scenes.

We will build a prototype OpenThread lightbulb following examples that are explained in several pieces of research we examine in the Implementation section. The rest of the literature review aims to shed light on the connection protocols that we have considered for our prototype. We compare Thread against ZigBee, Bluetooth, and WiFi. We discuss several metrics that are vital to the success of an IoT system in large-scale industrial spaces, covering factors such as power consumption, performance, scalability, and security.

The following subsection details the deliverables for the Spring 2024 semester; future improvements may seek to upscale the products for application in large-scale aquariums, aviaries, etc.

2.1 Deliverables

2.1.1 Prototype light

CS388 Methods for Research and Dissemination, November 2023, Earlham College

The prototype is a Thread-capable device that mocks the functionalities of a smart light. The light consists of groups of diodes categorized by their colours, i.e. UVs, Violets, Blues, Reds, and Greens. The light's only intrinsic attributes are the on/off states and power levels of each group of diodes.

2.1.2 *iOS App*

The App is the frontend through which users control the lights. We expect the app to support the following features, explained in the chronological order of a typical usage scenario:

- Adding Thread devices to HomeKit. The devices then show up in and can be controlled through both the App and HomeKit. This is a common feature among smart home apps.
- Inspecting device attributes. For this project, the app uses the the light's intrinsic attributes to calculate advanced attributes: color temperature, spectrum, and par level. No existing smart home platforms or smart aquarium products show these advanced attributes without additional professional equipment. The goal is to help aquarists make informed decisions on their lighting configuration instead of having to rely on optical perception which often fails to be accurate even for the most experienced aquarists.
- Saving predefined attribute settings as Scenes and exposing them to HomeKit. For example, a typical midday reef Scene could have the light's overall intensity set to 50%, with UVs, Violets, and Blues at 80% (80% of overall 50%, meaning they are actually set to 40%), and Reds and Greens at 25%. Users can set any supporting light to this Scene with a single button press. HomeKit does not support non-standard device attributes. This means the only way to control niche attributes that are only applicable to aquariums and aviaries is to abstract them into predefined Scenes [16].
- Scheduling daily profiles. For example, users may want to turn on the Morning Scene at 08:00, followed by the Midday Scene at 12:00, then the Late Afternoon Scene at 16:00. Scheduling enables automation of aquarium maintenance and, in the case of lighting, approximation of sunlight changes throughout the day on a natural reef.

2.1.3 *Integration APIs and Guidelines*

This is a wrapper of features applicable to pet care in OpenThread to enable fast integration of third party Thread devices. It defines the hardware attributes of Thread devices and how these attributes translate into frontend features. This streamlines third-party integration because manufacturers now only need to implement the intrinsic settings of a device to take full advantage of the platform's advanced smart settings for that device type. At the end of the project, the APIs will support both reef lighting and aviary lighting. Additionally, should we have time, we may explore device types such as wave-makers and protein skimmers, which are essential reefing equipment that should benefit a great deal from integration into a smart home platform.

3 Related Work

3.1 Homebridge

HomeBridge is a lightweight third-party solution that emulates the iOS HomeKit API to enable HomeKit integration for smart devices manufactured for other smart home platforms [7]. The HomeBridge server can be run on a Raspberry Pi, where it interprets the device's native IoT calls (like Google Nest) into HomeKit calls. HomeBridge allows abstraction of device features that are not available in HomeKit into things like Scenes or even disguise them as a similar feature. For example, wave-makers are a niche device category that is not supported in HomeKit, but there is a category for fans. The typical HomeBridge solution is to represent a wave-maker as a fan so we can control its rotation speed in HomeKit.

While Homebridge is not a replacement for OpenThread, the connectivity protocol upon which we will build our prototype, its codebase explores ways to emulate the HomeKit environment and is an excellent example of bespoke HomeKit integration. More importantly, we will reference Homebridge's implementation guide [8] to build the part of our iOS app that abstracts in-app functionalities into Scenes.

3.2 OpenThread

We have examined multiple studies that created OpenThread prototypes to conduct performance evaluation of the Thread protocol against other popular protocols such as ZigBee, Bluetooth, and WiFi. Ultimately, Thread emerges as the superior choice, owing to its remarkable combination of low latency, minimal power consumption, and the strategic integration of Border Routers, devices facilitating seamless bidirectional connectivity between the Thread Mesh and the internet [3] [4]. The inclusion of Border Routers greatly enhances the dependability of a smart home platform by incorporating multiple safeguards into the system. This effectively eliminates the likelihood of most instances of device failures leading to a complete shutdown of the entire system.

4 Implementation

The most common implementation of Thread is OpenThread, an open source project headed by Google. The OpenThread documentation details how to simulate a Thread network with minimal hardware implementation [2], through which we will explore the Thread topology and tinker with HomeKit integration.

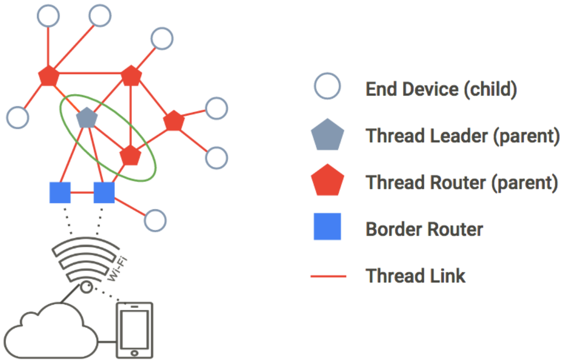

Figure 1: Thread Network Topology [2]

Following examples set by Khattak et al. [11] and OpenThread's documentation, we examined how to create a prototype device to test and showcase the scalability of our APIs using the Raspberry Pi, the nRF52840 dongle, and OpenThread 1.2. The research also provides recommendations as to which communication system (one-hop, two-hop, and multi-hop topologies) to utilize for specific usage scenarios. In the case of this project, which aims to create the basic framework for building comprehensive pet care IoT systems, the multi-hop topology supports the most flexible deployment.

Hardware-wise, Kim et al. [12] offers an in-depth look at the process of setting up a smart light bulb through the Silicon Labs EFR32 development package [13]. A detailed discussion among HomeKit developers [18] notes that the package offers the most comprehensive Thread development tools. The package comes with an established SDK that will help shorten development time.

5 Budget

Table 1: Budget

Hardware	Role	Price
HomePod mini*	Thread/HomeKit Hub	US$100
EFR32 Starter Kit	Wireless SoCs & SDK	US$200 [15] [14]

6 Timeline

The project will be implemented according to the timeline in Table 2.

Table 2: Project Timeline

Date	Milestone
Winter Break	Study OpenThread/HomeBridge implementations
	Purchase required hardware
Jan 22 - Feb 04	Build prototype with OpenThread
Feb 05 - 18	Build iOS app and integration APIs
Feb 26 - Mar 10	Debugging and Optimisation
	First draft of paper
Mar 11 - Mar 24	Second draft of paper, first draft of poster
Mar 15 - Apr 14	Revision of paper and poster

* This will determine what data is being exposed to HomeKit

7 Acknowledgement

I would like to thank Dr. David Barbella for his support and feedback on my project ideas and research proposal.

References

[1] Sarah A. Al-Qaseemi, Hajer A. Almulhim, Maria F. Almulhim, and Saqib Rasool Chaudhry. 2016. IoT architecture challenges and issues: Lack of standardization. In *2016 Future Technologies Conference (FTC)*. 731–738. https://doi.org/10.1109/FTC.2016.7821686

[2] Jeff Bumgardner. August 14, 2023. Simulating a Thread network with OpenThread. Retrieved October 18, 2023 from https://openthread.io/codelabs/openthread-simulation-posix#3

[3] Jingwei Dai. 2023. Literature Review: A Comparison between Thread and Other IoT Protocols.

[4] Kangping Dong. August, 2023. Thread Border Router - Bidirectional IPv6 Connectivity and DNS-Based Service Discovery. Retrieved October 16, 2023 from https://openthread.io/codelabs/openthread-border-router#0

[5] Simon Hill. 2022. Here's What the 'Matter' Smart Home Standard Is All About. Retrieved October 9, 2023 from https://www.wired.com/story/what-is-matter

[6] Signify Holding. 2023. Zigbee 3.0 support in Hue ecosystem. Retrieved October 15, 2023 from https://developers.meethue.com/zigbee-3-0-support-in-hue-ecosystem

[7] Homebridge. 2023. Homebridge. Retrieved December 1, 2023 from https://github.com/homebridge/homebridge

[8] Homebridge. 2023. Homebridge Plugin Development. Retrieved December 1, 2023 from https://developers.homebridge.io/#/

[9] Apple Inc. 2022. Matter support in iOS 16. Retrieved October 15, 2023 from https://developer.apple.com/apple-home/matter/

[10] Apple Inc. 2023. Apple unveils iPhone 15 Pro and iPhone 15 Pro Max. Retrieved October 15, 2023 from https://www.apple.com/newsroom/2023/09/apple-unveils-iphone-15-pro-and-iphone-15-pro-max/

[11] Sohaib Bin Altaf Khattak, Moustafa M. Nasralla, Haleem Farman, and Nikumani Choudhury. 2023. Performance Evaluation of an IEEE 802.15.4-Based Thread Network for Efficient Internet of Things Communications in Smart Cities. *Applied Sciences* 13, 13 (2023). https://doi.org/10.3390/app13137745

[12] Hyung-Sin Kim, Sam Kumar, and David E. Culler. 2019. Thread/OpenThread: A Compromise in Low-Power Wireless Multihop Network Architecture for the Internet of Things. *IEEE Communications Magazine* 57, 7 (2019), 55–61. https://doi.org/10.1109/MCOM.2019.1800788

[13] Silicon Laboratories. 2023. Silicon Labs EFR32 Wireless Gecko Technology Features. Retrieved December 1, 2023 from https://www.silabs.com/wireless/technology

[14] Silicon Laboratories. 2023. SLWRB4180B EFR32xG21A Wireless Gecko 2.4 GHz +20 dBm Radio Board. Retrieved December 1, 2023 from https://www.silabs.com/development-tools/wireless/slwrb4180b-efr32xg21-wireless-gecko-radio-board

[15] Silicon Laboratories. 2023. SLWSTK6021A EFR32xG22 Wireless Gecko Starter Kit. Retrieved December 1, 2023 from https://www.silabs.com/development-tools/wireless/efr32xg22-wireless-starter-kit

[16] Nanoleaf. 2021. iOS Scenes FAQ. Retrieved November 29, 2023 from https://helpdesk.nanoleaf.me/en-US/ios-scenes-faq-15695

[17] S. Pradeep, T. Kousalya, K.M. Aarsha Suresh, and Jebin Edwin. 2016. IoT and its connectivity challenges in smart home. *International Research Journal of Engineering and Technology (IRJET)* 3 (2016), 1040–1043.

[18] u/coyotanark. March 28, 2021. DIY OpenThread. + HomeKit. Retrieved November 29, 2023 from https://www.reddit.com/r/HomeKit/comments/mfc0u1/diy_openthread_homekit/