CS Capstone Technology Review

November 16, 2017

Project LOOM — Design an Internet of Things Rapid Prototyping System.

Prepared for

Chet Udell

Prepared by

Group 36

Project LOOM

William Selbie

Abstract

This document goes over the general goal of the project as well as my role in accomplishing this goal. It then proceeds to outline and compare the various technological solutions that can be implemented to achieve the goal of the project.

CONTENTS

1 INTRODUCTION

1.1 Role

I am on Project LOOM's Firmware team along with Trevor Swope. The role of the firmware team is to write all of the sketches that will be run on the LOOM kits' hardware. The sketches must be written as to allow for great modularity so that the Project LOOM provided graphical user interface can flash the hardware with the most minimal sketch that can still achieve what the user wants.

1.2 Goal

The aim of Project LOOM is to design kits of hardware that require minimal assembly and have explicitly stated functionality. These kits can then be programmed by using a graphical user interface that gives the user control over the inputs, data processing, and outputs of the system. I specifically will focus on the what network platforms and architecture can best be used to accomplish the Internet of Things aspect of Project LOOM, as well as the best way to maintain good version control on a project that will have multiple people adding, modifying, and removing parts of the code base at any given time.

2 NETWORK ARCHITECTURE

After having its firmware programmed, each device in a Project LOOM kit is going to be capable of at least two of the following functions:

1) Gathering data from a sensor
2) Performing an action with an actuator
3) Transmitting data to another device on the network
4) Receiving data from another device on the network
5) Processing gathered or received data
6) Transmitting data to a device connected to the world wide web

However, not all of the devices will need to perform each of these functions, so it is necessary to decide how the networks will be architected to make the most efficient use of the kits' resources.

2.1 Distributed

The first option for the network architecture is to have each device be capable of all of these things. Each device in the network would then gather data, process that data, and then transmit that data to a device connected to the world wide web. The devices connected to sensors would also contain the logic to transmit commands to devices connected to actuators when the data being input met certain criteria. The benefit of this would be that the devices could all be programmed in quite similar ways, and there would be no central component which would stop the whole system if it broke. While this is the most robust system, it is also the most bloated, a system in which only one device is capable of processing data and relaying it to the outside world could accomplish the same functionality with less overhead. One of Project LOOM's principal goals is that kits have long lifespans (meaning minimal battery usage) and require little maintainence. An architecture like this would also drive up the price of the kits as each chip would need great storage, battery life, and processing power.

2.2 Gateway Centralized

The second option is to have a Gateway Centralized architecture, wherein each device is either a node or a gateway. Nodes are capable of gathering and transmitting data, or actuating and receiving data. Nodes gathering data will periodically transmit data to a the network's gateway. The gateway will process the data, and if certain criteria are met, send commands to the nodes that have actuators. The gateway will also periodically transmit the received and processed data to a globally connected device. This architecture uses more specialization that the previous, and as such each device will be leaner, as it will be required to do less. The result of this will be a cheaper kit that lasts longer. However, in the event that the gateway does break, the whole system will stop working, which makes it a more fragile solution than the Distributed network. It also limits the size of the kits and individual networks, because one gateway will only be able to support a finite number of nodes.

2.3 Gateway Centralized with Cloud Service

The third option is to create a gateway centralized network, with the single difference that the gateway does not perform any data processing. Rather, it transmits all data to cloud computing service that will then process the data and distribute it to its end location. The benefit of this would be to make the gateway leaner by requiring to do less, with the downside being that it adds another breakable part in the process. In projects where it would be necessary to measure data for extended periods of time, this would be an ideal choice because it would provice the minimum functionality needed to do so. However, for projects requiring extensive use of actuators, then the gateway would need to do some processing, or be capable of receiving data from a globally connected device. This would add more pieces to the system than necessary. This would also cause kits to cost more in order to pay for the cloud computing services, and may not be as extensible as the other network architectures because of it.

Overall, the Gateway Centralized architecture appears to be the best choice as it maintains as much flexibility as possible in usage, while allowing for the use of cost effective chips making up specialized devices.

3 Inter-Device Communication Platform

Project LOOM has several options to choose from for how the devices in a network will talk to one another, each with their own benefits and drawbacks.

3.1 WiFi

The most obvious option for wireless inter-device communication is of course WiFi, more specifically, a wireless LAN (local area network). WLANs have a high bandwidth and bit-rate compared to other wireless standards, which is at the cost of greater power consumption. The range for wifi typically ranges from 150ft inside to 300ft outside, though newer devices can achieve more [2]. A WLAN configuration would ensure that each device has the bandwidth needed to transmit all of the data that it gathers, however, it also means shorter battery life spans for each of the devices. Furthermore, it means that each device will have to fall within 300ft of whichever device is connected to a global network.

3.2 LoRa

LoRa is an LPWAN (Low Power Wide Area Network), meaning that the range of the network is significantly larger than a WLAN, in the range of 3km in urban areas up to 30km in rural areas [1]. Furthermore, the power consumption is also significantly lower, which is at the cost of an equally significant loss in bit-rate. Chips with LoRa are similar enough in cost to WiFi enabled chips that it's not of significance. The choice between LoRa and WiFi revolves entirely about how much data each device needs to send, because if it is only a small amount then LoRa is the clear choice, due to the longer lifespan that it guarantees for the devices' batteries. LoRa also allows for networks of a much greater geographical span that WiFi, which would make projects that require the measurement of attributes on a large scale much easier.

3.3 Bluetooth

The last consideration for Inter-Device Communication Platform is Bluetooth, a short-range communication method typically used between mobile devices. Bluetooth is shorter range than WiFi and LoRa, offering less bitrate than WiFi, but more than LoRa. Bluetooth chips are only slightly more expensive that WiFi and LoRa chips, so price will not be the deciding factor. Bluetooth's short range means that the network will only be able to contain devices in a small range, up to 100ft [3]. Furthermore, Bluetooth is not designed like WiFi is to have a central hub or gateway be used in communication, rather it is typically a connection used between two specific devices. This means that architecturally, it is also less likely to suit Project LOOM's needs.

Both WiFi and LoRa are strong contenders for use in Project LOOM, however, given that most devices plan only to transmit raw data, it is unlikely that any device would ever take full advantage of the higher data transmission rates that WiFi offers. Rather, the devices are much more likely to make use of the longer lifespan that's guaranteed by LoRa, making LoRa the most apt choice.

4 Version Control

Project LOOM will require the creation and maintenance of a large code base that must be easy for more experienced users of Project LOOM's kits to obtain and modify to suit their own needs. Furthermore, Project LOOM's own developers need the ability to each work on distinct updates and additions, all the while being able to seemlessly merge these new pieces of code with the central repository.

4.1 Git

Git is an open-source distributed version control system that allows for the easy development of multiple distinct additions and updates simultaneously. Git's robust branching system can be used by developers to work on multiple updates of their own locally, all while other developers are doing the exact same thing. Project LOOM's codebase will have multiple developers working on the creation of multiple features at any given time, and the ability for these developers to work separately but still smoothly merge their changes with the main repository is absolutely crucial. Furthermore, Git's method of tracking vesions through hashed differences means that the history and path taken by Project LOOM will be concretely held, making documentation of the codebase as a whole significantly easier.

4.2 Mercurial

The other major distributed version control system is called Mercurial. Mercurial's approach to the documentation of changes between versions is much slimmer than Git's, resulting in documentation that makes it easier to find what you are looking for. Beyond this, Mercurial is also a lot friendlier than Git. Git has a Linux oriented style of doing things where the command line is Git's interface, and it's documentation is man-page like in its density. A steeper learning curve may result in more hours being spent figuring out Git and fewer actually writing code for Project LOOM. In contrast, Mercurial is much more like Windows in how it boasts several projects providing a GUI for Mercurial, as well as offering professional support and training should it be needed. While the ease of learning a piece of software should not be a huge concern of Project LOOM's developers, nonetheless it is important to consider for the sake of the experienced users that hope to modify Project LOOM's code for their own ends, and will have to use Project LOOM's chosen version control system to do so.

4.3 Subversion

Another option for version control is Subversion, which, unlike Git and Mercurial, is a centralized version control system. Subversion is a much more straightforward approach to version control because wherein rather than keeping a local copy of the repository that you can change any way you please, you simply decide which file you would like to change, and then lock it if you wish to ensure that no one else can modify at the same time you are. This ensures that you never have issues merging your changes with the main repository because another develope made changes to the file that you made changes to. One issue with Subversion is that not having a local copy of the repository will cause issues for Project LOOM developers if they woudd ever like to view or edit code that they don't have checked out when they are not connected to the internet. This is a minor concern though as it is unlikely that such a situation will occur.

Of all of these, Git appears to be the most attractive option as its branching and history keeping are almost tailored for Project LOOM's needs. While Mercurial is certainly much more user friendly, the average user that decides to modify Project LOOM's code is most likely savvy enough to figure out a thoroughly, though densely, documented version control system.

References

[1] What is lora? http://www.semtech.com/wireless-rf/internet-of-things/what-is-lora/. Accessed: 11-06-2017.

[2] Bradley Mitchell. The range of a typical wifi network. https://www.lifewire.com/range-of-typical-wifi-network-816564. Accessed: 11-06-2017.

[3] Joshua Wright. Dispelling common bluetooth misconceptions. https://www.sans.edu/cyber-research/security-laboratory/article/bluetooth. Accessed: 11-06-2017.