Virtual Node Environment for the Waggle Attentive Sensing Platform

Saikiran Yerraguntla

Argonne National Laboratory: Mathematics and Computer Science Division
Bachelors in Computer Science and Applied Mathematics, Illinois Institute of Technology
syerrag1@hawk.iit.edu

***Supervisors**: Pete Beckman, Raj Sankaran, William Catino, Peter Lane, Sean Shakarami*

ABSTRACT

Waggle is an attentive sensing platform that brings together an ensemble of physically distributed Internet enabled sensing and computing devices, and a cloud enabled storage, control and data dissemination infrastructure. However, research and development on the platform using physical hardware nodes can be expensive, time-consuming and tough to scale. Vital objectives such as sensor plugin development, automated software testing, and stress-testing the cloud resources can be enabled by creating virtual Waggle sensor nodes. The environment that allows for the creation and deployment of a limitless number of virtual nodes is the Virtual Node Environment. The applications used to create and deploy these virtual nodes were RabbitMQ and Docker. While RabbitMQ laid the data pipeline for setting up node and server-side plugin clients, Docker was used to create the virtual node and its environment as containers. The containers will later be used to virtually run software plugins, automate software testing, lay a basis for further creation of more virtual nodes, and to create a swarm of a multitude of nodes to stress-test the Waggle server "**Beehive"** for further development.

Keywords: Virtual node, virtual, deployment, RabbitMQ, Docker, development

INTRODUCTION

A. Project Mission:

Waggle is a research project at Argonne National Laboratory designated to design, develop, and deploy a novel wireless sensor platform to enable a new breed of sensor-driven environmental science and smart city research. The Waggle platform architecture uses emerging technology in low-power processors, sensors, and cloud computing to build powerful and reliable sensor nodes that can actively analyze and respond to data.

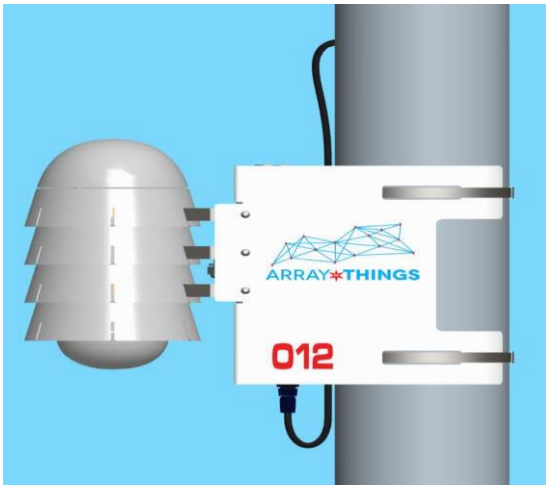

Figure 1. AOT sensor node

Developing automated software for testing the sensors, stress-testing Waggle's server "**Beehive"**, and improving the architecture of the AOT nodes require using actual hardware nodes as shown in Figure 1. However, research and development on these physical nodes can be expensive, time-consuming, and tough to scale. Thus, virtual nodes were created to enable advancement in the Waggle platform, to allow quick, superior qualitative testing and deployment of the sensors and nodes. The environment that allows the creation and deployment of an infinite number of virtual nodes and enables the user to explore the inside of a node through a medium of their own machine, is the Virtual Node Environment.

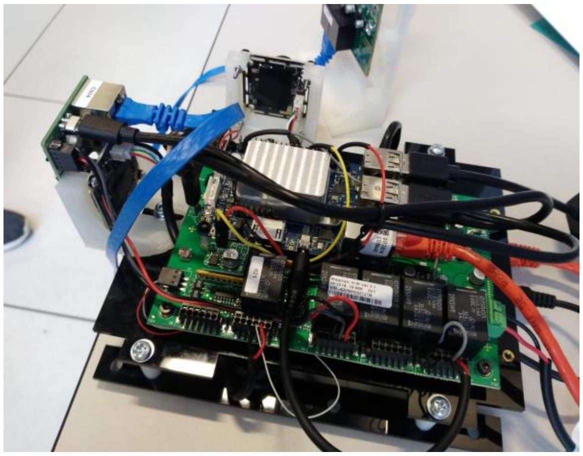

Figure 2. Compute brain of AOT/WAGGLE node

Figure 3. Machine running multiple virtual nodes internally, as seen inside computer

B. Project Scope Overview:

In order to test the sensors and other node hardware virtually, a data-pipeline was laid out to send all the software test results to the Waggle server, “Beehive”. From Beehive, the data is aggregated and analyzed for further improvement. The application used to build the foundation of the pipeline is RabbitMQ.
Subsequently, containers were created to virtually replicate the action and performance of a hard node. These containers served as our virtual nodes and the application used to create these containers was Docker. The nodes led to the creation of the Virtual Node Environment.

Future implications for this project include developing software infrastructure that will create micro-services to deploy virtual nodes in numerous containers, and creating a QEMU emulator for node emulation.

I. PROBLEM AND PROPOSED SOLUTION

Research and development on the Waggle platform using physical hardware nodes can be expensive, time-consuming, and tough to scale. The proposed solution was to deploy virtual nodes that would facilitate towards the improving and advancing the project.

The virtual nodes will later cater towards executing plugins for sensor testing and server enhancement. Ultimately, the hardware and software can be tested and deployed to the actual node immediately. These nodes will then be revised and installed around the Chicago area by the Array of Things team.

II. METHODOLOGY

Before creating the virtual nodes, a data -pipeline had to be built to send any test data from the software analysis to the Waggle server. Thus, the project was split into 2 parts namely: Creating the Plugin Data-Pipeline and Creating the Virtual Node Environment.

A. Creating the Plugin Data-Pipeline:

In order to test certain software via virtual nodes and observe how the data is parsed through the server and output to a web-interface, various plugin clients must be set up to process the data formulated by the software. This will allow the developers to access their test data from the software and hardware analysis they performed on the Waggle platform, and understand what modifications must be done to their program application in order to achieve tangible results. This will also provide them opportunities to fiddle and alter various parameters of their plugins to see their desired results.

The data-pipeline was built using applications called RabbitMQ and python scripts. RabbitMQ is a broker messaging exchange service that facilitates maintaining, handling and processing data in the form of packets/messages from one end to another. In this case, the pipeline will transmit packets of data from sensors, hardware devices, cloud-based databases or from a local machine, and publish to the RabbitMQ broker system. Once the data is sent to the broker, the messages/packets are then maintained, processed, and sent, one-by-one, to Waggle’s Beehive server.

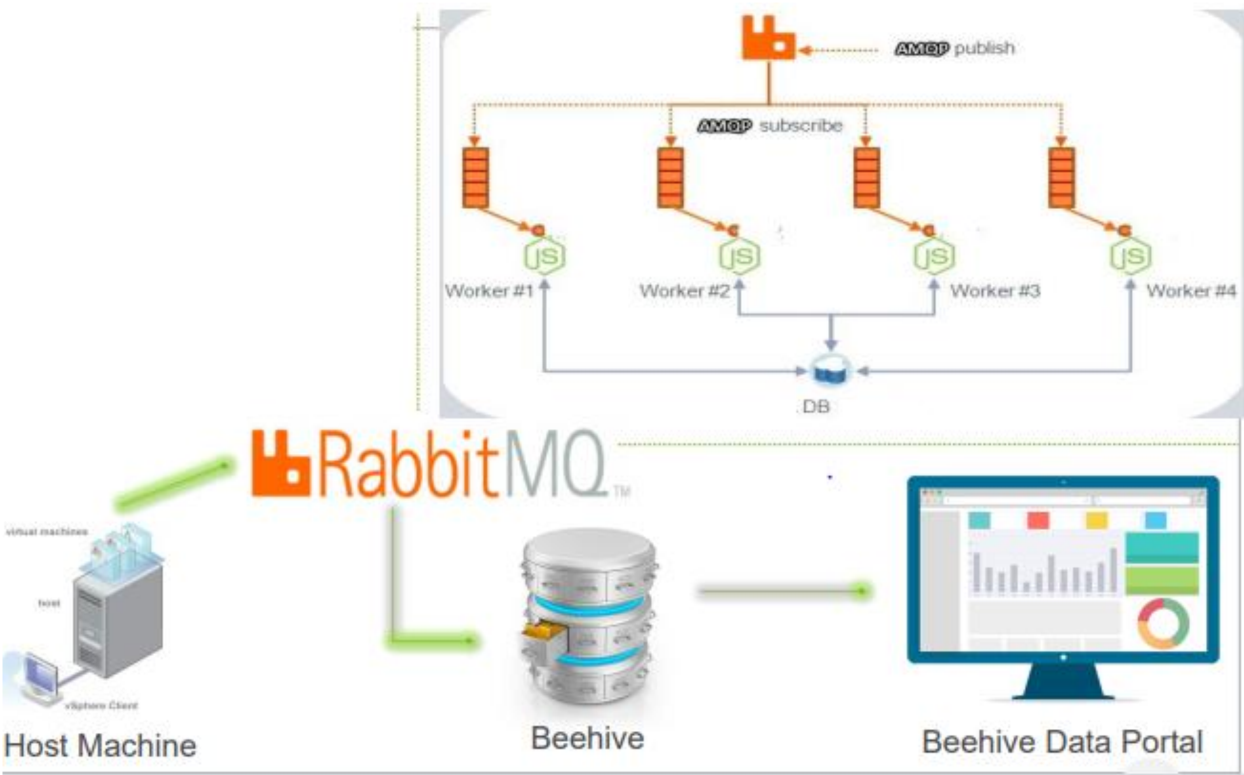

Figure 4. The arrows designate the data-pipeline that was built about the Waggle platform for plugin software testing purposes

Once the data-pipeline was established, plugins were created in order to allow the developers to send their test data in different form. The plugins the developer must implement are:

1. ***Node-side plugin:***
   The Node-side Plugin queries sensors, cloud-platforms, and other data sources. Further, it decodes raw data and sends it to Waggle's cloud database platform, Beehive, to be published in Beehive's Web Data Portal.

2. ***Server-side plugin:***
   The Server-side Plugin, situated in Beehive, retrieves the raw data from the node-side plugin, processes it, and publishes the output to Beehive's Web Data Portal.

Beehive has a web data portal, created from a flask app, that collects the aggregate of data from the plugins, and uploads it to the website in the form of csv files. The web data portal is split into two columns. The first column populates raw data from the node-side plugin. The second column populates the processed decoded data from the server-side plugin. This allows the developer to download and view the csv files from the web data portal, to ultimately understand the results from their analysis.

An example of the application was testing thermistor sensors used in the sensors nodes. These sensors send a temperature reading every five seconds. A node-side plugin was created to publish the raw temperature readings to the broker service, which sent packets of data to the web data portal's raw database column. A server-side plugin was created to plot the temperature readings from the sensor, and publish the graph to the web data portal's decoded database column.

B. Creating the Virtual Node Environment:

Once the data-pipeline was ingrained to the platform, virtual nodes were than created using an open-source application called: "Docker". The virtual nodes were programmed to mirror the exact functioning and purpose as of that the actual hard node. This allowed the Waggle team to use these containers to test their software and hardware on the Waggle platform using the virtual nodes as proxies for software plugins. These containers publish data to the server's web data portal as outlined in Figure 4.

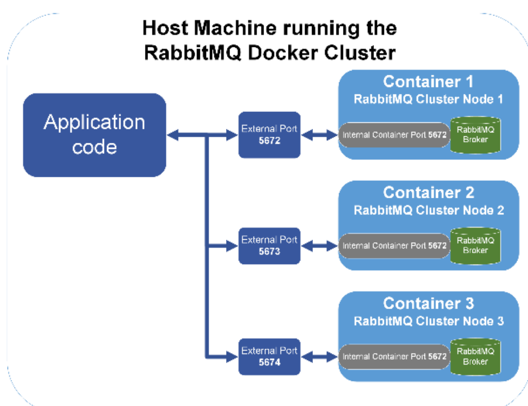

Figure 5. The illustration demonstrates containers interacting with RabbitMQ to send data from within the virtual to the Beehive server.

An infinite number of containers can be replicated and set to run numerous instances of virtual nodes, ultimately stress-testing the Waggle's server, "Beehive", constantly. This will enable us to understand the defects of the server and enhance it to its maximum efficiency. Likewise, the user or developer can enter into the virtual node and explore the libraries, dependencies, and all the other software the virtual node uses to operate. Consequently, this helps us understand how the brain and other processers of the node can be improved upon for better performance.

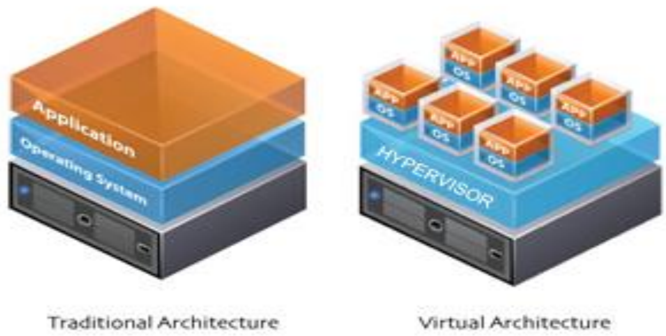

Figure 6. A multitude of virtual nodes can be created on a single Linux box for stress-testing the server, as seen in the figure to the right

One such application used for the testing purposes of these containers was being able to virtually run "Coresense Chips" (A vital processor of the sensor-node) and "Thermistor sensors" (One of the sensors of the sensor-node). From this experiment, we observed how the containers processed the data coming from these data hubs and sent the data to the Beehive's web data portal using the data-pipeline formerly created.

III. CONCLUSION

The creation and establishment of these virtual nodes has laid a foundation to set proxies to basis to create more virtual nodes, and automate software and hardware testing. The goal of the project was to create a sub-platform within the Waggle platform, for primarily qualitative testing and software analysis of the Waggle technology.

Further directions of the project lead to developing further software infrastructure that will create microservices to deploy virtual nodes in numerous containers and creating QEMU (Quick Emulator) for Node Emulation.

ACKNOWLEDGEMENTS

I wish to thank The United States Department of Energy, Argonne National Laboratory, and the National Science Foundation for offering me this prestigious internship to work with a dedicated team of developers towards building a project in hopes of bettering our society and environment. I would like to thank Waggle for allowing me to work with them to accomplish their goals. I would also like to thank the Array of Things team for taking their time and effort to deploying the sensor-nodes around the Chicago area.
Finally, I would like to thank my mentors: Dr. Pete Beckman, Dr. Raj Sankaran, Dr. William Catino, Dr. Peter Lane, Dr. Sean Shakarami and the rest of the Waggle team for providing me the support and guidance required to tackle all complications and achieve my mission which would later serve as a greater purpose for the project in the coming years.