Deploying Edge Computing to Augment Endpoint Functionality

Ciprian Popoviciu, Philip Lunsford, John Pickard, Colby Sawyer, Sean Wears, Spencer Lee, Zachary Zynda

Abstract: The seamless integration of additional endpoint processing and storage as a separate component to an endpoint appliance can provide a platform for additional capabilities and services to existing products and services. A multidisciplinary project is presented where students were given high level objectives to add additional compute and networking functionality to field deployed mobile Cisco WebEx Boards. These additional resources support service monitoring and troubleshooting tools by providing a separate user agent to continually test the end-to-end network and control plane network performance. Students from BS in Industrial Engineering Technology, BS in Design, BS in Information and Computer Technology, and BS in Computer Science worked together to complete the project and deploy integrated Raspberry Pi systems to Cisco WebEx boards funded by USDA for community college and high school collaboration and distance education. Tasks accomplished by the student team include physical design, fabrication, installation, configuration, instrumentation development, provisioning and deployment. An assessment of student learning outcomes unique to the interdisciplinary project will be presented. This plug-and-play solution provides a platform for future innovation by adding functionality such as intrusion detection, user logging, inventory tracking, and hardware diagnostics.

Key words: IP, performance, Cisco WebEx Board, Raspberry PI, Industrial Design, SaaS, Monitoring.

1. Introduction

Modern IT environments are driven by two major dynamics: the migration of services to cloud based delivery [1] and an exponential growth of end-point devices [2] enabling or supporting services. These two trends have a major impact on the architecture, structure and operation of current and future cyberinfrastructures. Specifically, the most significant impact relates to great portions of the network becoming, in the view of the services being delivered, simply transport commodity. The “intelligence” of the infrastructure is migrating towards the cloud where services are controlled or towards the edge where data can be processed in a more local, less expensive way [3]. These changes require new technologies being deployed, new skills for the staff and new operating processes. This paradigm change applies to all cyberinfrastructures including high education institutions.

The march to the Cloud is well accepted and well on its way for many organizations. The rise of edge computing on the other hand is a more recent trend, naturally lagging the centralization of service delivery control. This means that for the next couple of years, cyberinfrastructures will be in a state of transition, a hybrid of traditional and new architectures [4]. Services will migrate to cloud lacking the operational tools for optimal delivery and management while edge computing requirements will be met by compute resources being bolted onto exiting infrastructures based on operational or service needs. This period of transition is a great opportunity for innovation in many areas of IT management and IT service delivery.

In this paper we capture the process of addressing these two trends in the context of a new service deployment in a higher education environment. We cover the steps taken by a diverse team of students from Industrial Engineering Technology and Design [5], Information and Computer Technology [6], and Computer Science [7] to complement a cloud-based video conferencing service with the edge computing resources needed to manage the service. The paper captures the student learning outcomes and the future work enabled by the platform that was created.

2. Problem Definition

Department of Technology Systems at East Carolina University (ECU) received a grant from US Department of Agriculture (USDA) [8] to purchase Cisco WebEx boards [9] and deploy them to 12 colleges and high schools across Eastern North Carolina to facilitate distance education (Figure 1).

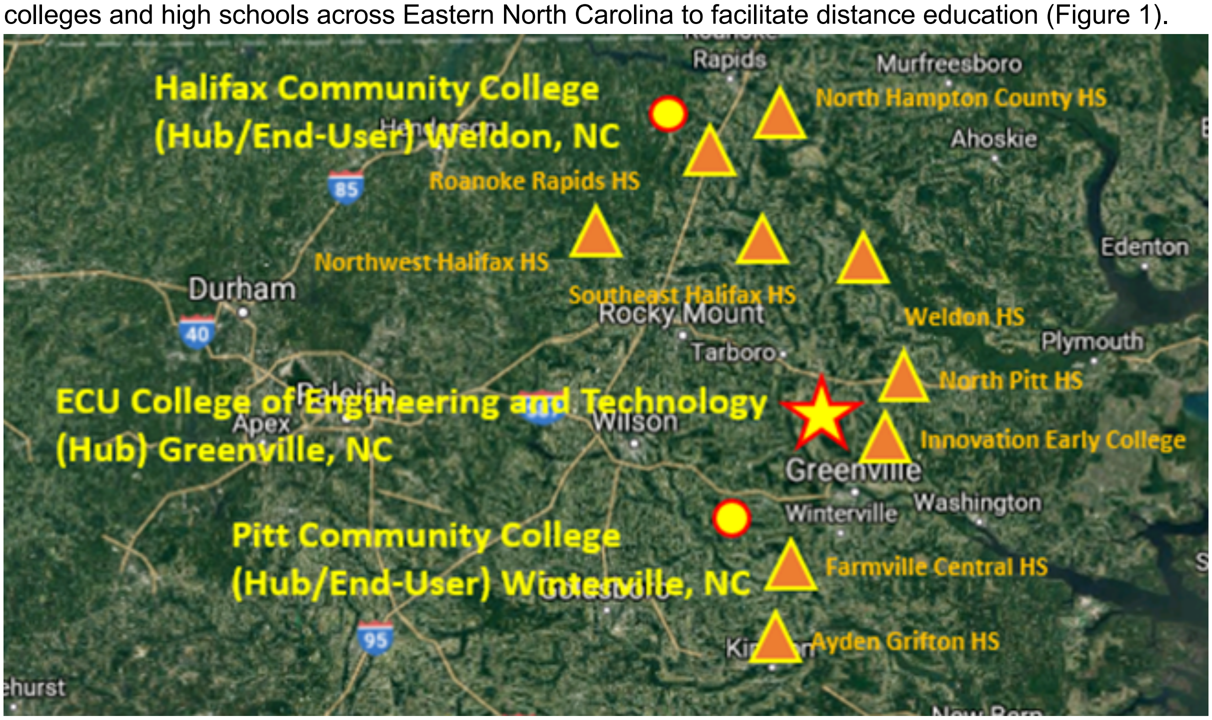

Figure 1. Footprint of the Planned Distance Education Service

This new service enables a more engaging, collaborative instructional experience involving students in multiple classrooms across Eastern North Carolina and ECU instructors. The potential positive impact of this service is significant. The USDA grant covers the cost of the hardware (Figure 2) and the WebEx cloud-based service provided by Cisco [10].

<include picture>

Figure 2. Cisco WebEx Boards mounted on mobile stands.

Cloud based services have been heavily marketed to give the impression that with the purchase of the endpoints and the service, the only thing needed to turn this offering into production is to assemble and deploy the WebEx boards. This is one of the reasons why organizations often time decide to skip working with their internal IT team to plan the rollout of cloud-based services. The practical aspects of operating IT services do however require answers to key operational questions:

- **How will these assets be track once they are deployed?** At a minimum, the service owner (ECU) should know if the endpoint is connected and what is the local network information
- **What is the quality of the control-plane path?** ECU and the partner organizations need to know if the WebEx service is available to set up the sessions
- **What is the quality of the data-plane path?** ECU and the partner organizations need to know if sessions established between various locations will deliver a good user experience
- **When there is an issue, who is responsible for it?** ECU and the partner organizations need to have the data that helps isolate fault domains and support any trouble-tickets opened with service providers.

Many of these questions might be answerable should ECU or the partner organizations have full management visibility in the WebEx service however, that is rarely the case. Ultimately, cloud-based services insert multiple management domains in the service delivery path: Local IT, Internet Service Provider (ISP), Internet Backbone and Cloud Service Provider (WebEx) as shown in Figure 3. This leads to significant uncertainty on which one of these domains owns the cause of an issue experienced by the service.

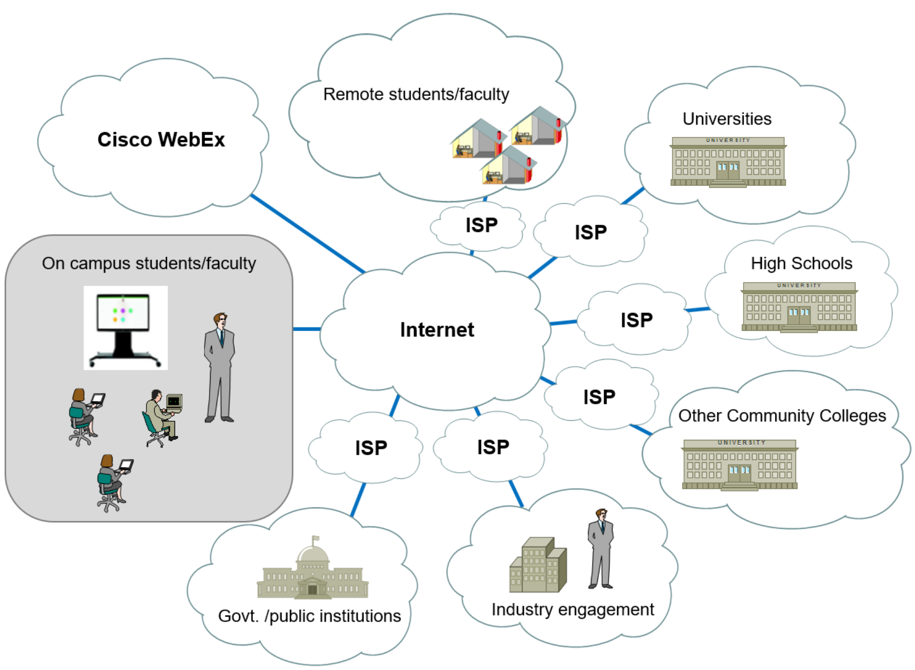

Figure 3. Management Domains for the WebEx Service.

This uncertainty becomes very problematic when issues crop up in service operation leading to significant downtime while finger-pointing delays resolution. Even if ECU explicitly states that it will not be responsible for the operational details of the service, it will inevitably find itself spending resources on issues that have nothing to do with its own IT environment. Moreover, ECU will have no access to the IT infrastructure hosting these endpoints thus limiting effective support.

The ECU team leading the service enablement project needed a solution that will enable it and the project stakeholders to have the visibility needed to manage the assets, the infrastructure and the service involved in this offering. The goal is to gain this visibility and enable self-governance without needing significant commitments from the IT teams at each of the locations.

3. Target Solution

To gain the information needed to enable self-management of the service without having to request local resources or support it was decided to add inexpensive, managed compute resources to the WebEx board stand. These compute resources can support the monitoring and troubleshooting capabilities needed to support the solution and to add additional capabilities in the future. Fundamentally, it was decided to add compute resources to the frames supporting the endpoints.

The edge computing platform enables ECU to collect relevant operational information for the service:

- **Network Attachment Information** – Collect information related to the access layer where the WebEx board is connected. IP address information, DNS information, WiFi metrics where applicable. This information can provide asset tracking information based on IP address assignment and/or geolocation services. This data can be collected assuming the EC platform is connected the same way as the WebEx board.
- **Location Availability Information** – Validate that the access layer at a given location is operational and available to the WebEx board
- **Network Path Quality** – Evaluate the quality of the network path between locations in terms of latency, drops and jitter
- **Service Availability Information** – Detect availability of WebEx service control plane at a given location.

Additional functionality will be added to the platform in the future.

The solution had to facilitate operations not to make the entire system more complex thus implementing specific capabilities:

- **Fit to Existing Hardware** – Build an enclosure that can be easily and securely attached to the existing stand
- **Same Layer Two Access Options** – Build the EC platform capable to offer similar access to the one used by the WebEx board (Gigabit Ethernet or WiFi)
- **Manageable** – Simple provisioning and management of the EC devices
- **Secure** – The EC platform should have a level of physical and access security that meets the standard requirements of educational institutions
- **Non-intrusive** – The tests executed by the platform should not negatively impact compliance. For example, the platform should not listen to traffic unless explicitly requested and allowed by the local IT organization

The subsequent sections detail the technology and design choices for this platform.

4. Compute Platform

The process of selecting the compute hardware for our EC platform took into considerations several factors:

- Capability to support targeted monitoring tools and services
- Support for both Gigabit Ethernet and WiFi (2.4 and 5 GHz)
- Off the shelf hardware with a large community and prove track record of continuous improvement
- Easy to develop on and to extend
- Low cost

The market has seen a surge in the number and types of embedded platforms that would fit such requirements and while various options were considered such as Beagle Bone Black [] and Intel Nook [] it was decided to go for the most supported and rapidly evolving Raspberry PI hardware. In particular, the recently released Raspberry PI 4[] was selected (Figure 4).

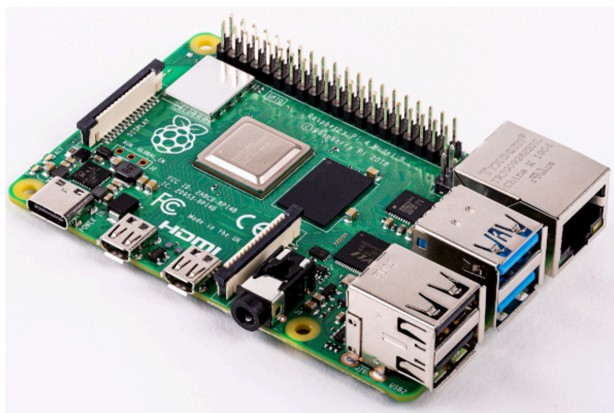

Figure 4. Raspberry PI 4 Hardware.

Raspberry PI 4 is featuring the following specifications relevant to the project []:

- CPU – Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- RAM – 1GB, 2GB or 4GB LPDDR4-2400 SDRAM (depending on model)
- WiFI – 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE
- Ethernet – Gigabit
- USB – 2 USB 3.0 ports; 2 USB 2.0 ports
- GPIO header – Raspberry Pi standard 40 pin
- Storage – Micro-SD card slot for loading operating system and data storage
- OS – Debian Linux 10 based

Despite the ARM architecture of the platform, Raspberry PI 4 has enough resources to support the needs of the monitoring tools planned for the initial release and future functionality such as security testing and local analytics. The 4GB RAM model has been selected with a minimum 16GB Micro-SD card with high read/write support. The Raspberry PI 4s used in deployment were fitted with heat sinks for the processor since it is documented that this version of PI is warming up more than previous versions. In some configurations of the EC device, the PI is fitted with a PoE (Power over Ethernet) hat.

Along with competitive pricing, another reason for selecting Raspberry PI hardware is the large community built around this device. This community provides significant support and vets the issues experience by such devices. The project team baselined the performance of the PI (see Section 6) independently of the data collected through research and feels comfortable with the current choice of hardware.

In order to accommodate Gigabit Ethernet access for the WebEx board and for the EC device while still using a single Gigabit port, a Gigabit Ethernet, PoE unmanaged switch was included as a requirement for the platform design. When a WebEx board connect to the network via Ethernet, it will connect through the EC device switch along with the Raspberry PI. In version 1.0 of the EC device a NETGEAR GS305-300PAS switch was used. In the meantime, Netgear changed its product line nomenclature and the switch is now identified as GS205 (http://www.downloads.netgear.com/files/GDC/datasheet/en/UnmanagedSwitches200.pdf?_ga=2.18132475.350375064.1580101840-719147796.1580101840). The modular design of the enclosure enables the project team to fit alternative hardware as the various components evolve.

Connectivity between the device components is accomplished with cabling selected for reduce bulk and high flexibility.

5. Enclosure Design

The enclosure design was based on the following design principles:

- **Observe Requirements**
    - **WiFi Access** – The need for Wi-Fi access required the housing to be plastic so 3D printing was the top option as molding parts would be cost and time prohibitive
    - **Mixed Access** – Fit both WiFi and Ethernet access for the hardware. Planning for the future when multiple Ethernet ports might be need for functionality such as port spanning [x], the enclosure had to fit a Gigabit unmanaged switch
    - **Modular** – The enclosure should support different hardware combinations dependent on the deployment model for the EC platform
    - **Flexible Mounting Point** – Develop an attachment specific to the WebEx board stand however, plan for alternate hardware platforms to witch the case might be attached in the future
    - **Resilient Wiring** – Wiring must be secured to avoid hardware damage caused by wire yanks
- **Three Times** – Everything should be designed at least three times to ensure things are not overly complex and they truly meet the design requirements
- **Keep it Simple** – If there are two or more competing designs, the simplest one is usually the best solution. With that in mind the first design iterations do not typically get saved but become an exercise in perspective and refocusing on what the part's true scope

The most time was invested into the attachment options. The enclosure had to be high up on the WebEx Board stand to avoid WiFi signal interference from the metal parts of the stand. For simplicity and ease of installation reasons, using existing features was the better solution. A ring and mounting fixture were designed through multiple iterations with the challenging part being the solution to securing the mounting. Based on efficiency, simplicity and safety considerations it was decided to route the wiring (power, ethernet) through the same opening used to mount the enclosure to the stand (Figure x).

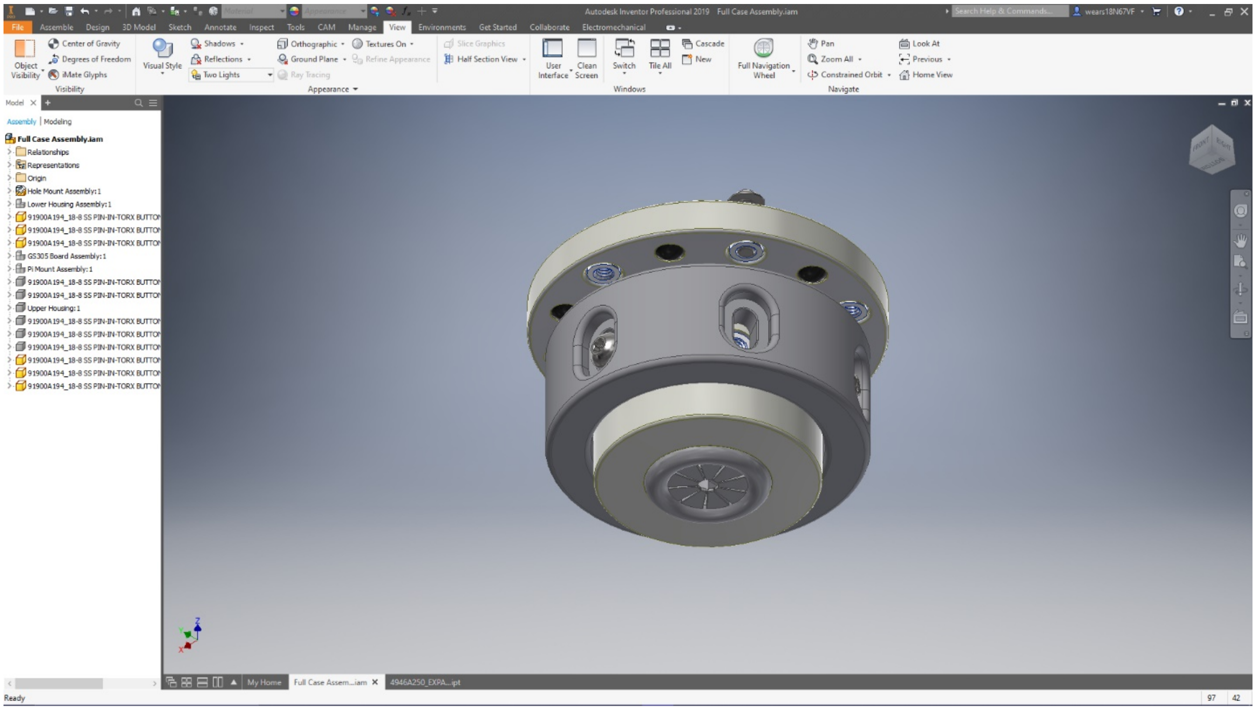

Figure x. Cisco WebEx Board Stand Attachment Point.

The design of the hardware housing started with simple components mounted to each other with the Raspberry PI mounted on top of the network switch to save space and reduce housing size. To reduce bending stress on the wires connecting the boards, we added large radii to each end of the housing. These radii also increased the overall strength of the housing over a rectangular design (Figure x).

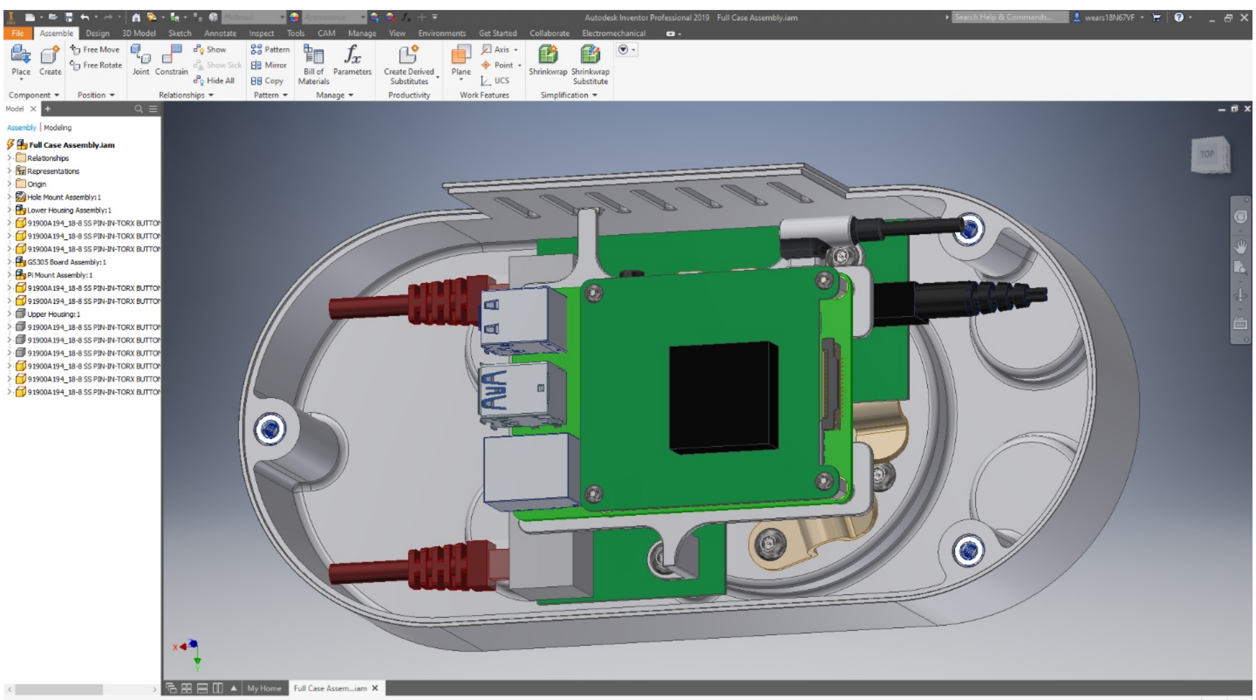

Figure x. Enclosure Design with Modular Hardware Assembly.

The Raspberry Pi 4 has symmetrical mounting holes and is well documented due to the open source nature of the device. The switch however did not have extensive documentation, so the design options were evaluated after a detailed analysis of the network switch board.

All the design work was done in Autodesk Inventor 2019 with CAD files of the mounting and other hardware imported via McMaster-Carr. The printing of the prototype was done on (insert make and model of 3D printer, need to verify with Andy) and (insert slicing software used) to get estimate print times and material usage to check against actual print time and material usage. The first version of the enclosure was printed with dissolvable support material to minimize post processing after the print finished. Appropriate tolerances were applied for part fitment, ABS part shrinkage and part finishing requirements inherent in the current 3D printing process. The assembly of the housing went smoothly with no major adjustments or design changes except for small artifacts from the printing process needed to be removed (Figure x).

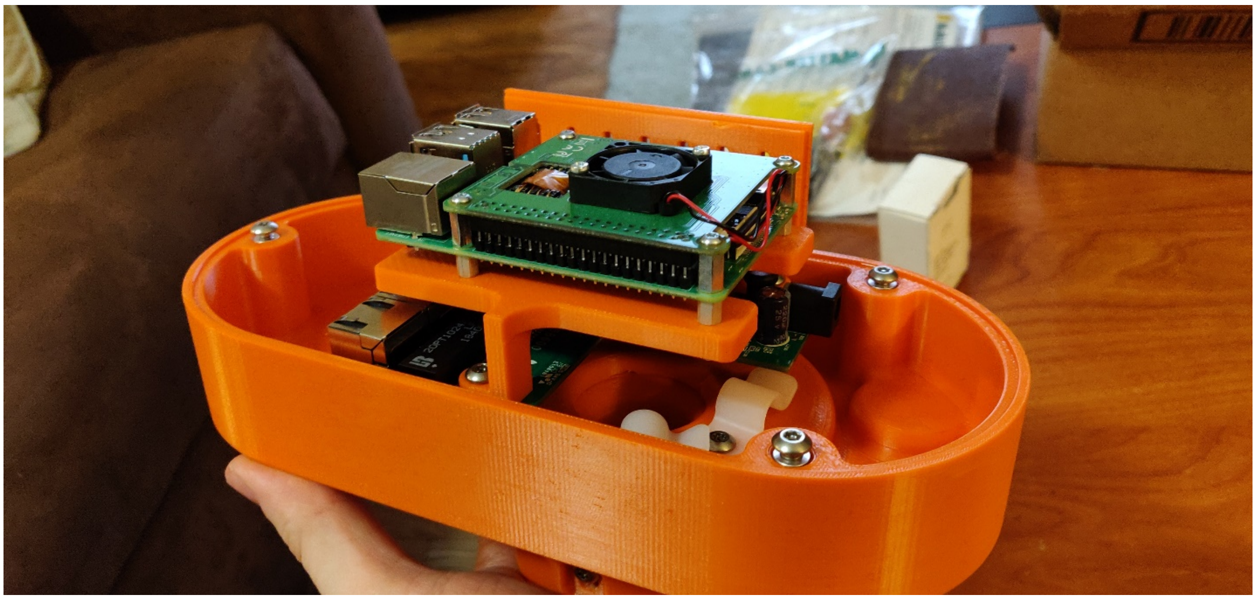

Figure x. Enclosure Design with Modular Hardware Assembly.

The enclosure securely attaches to the WebEx board stand with cabling secured according to the requirements (Figure x).


Figure x. Enclosure Mounted to the WebEx Board Stand.

The bill of materials for the first version of the fully loaded device (including Raspberry PI 4) is shown in Table 1. The per device cost will vary depending on the internal hardware configuration.

Pi Project Case Parts and Tools								
	Totals:	Price Per Assembly		Number of Assemblies		Total Cost		
		$ 139.15		1		**$ 206.74**		
Tools & Hardware								
Part Name	Part Number	Quant Per Pack	Price per Pack	Quant Per Assembly	Price Per Part	Price Per Assembly	Quant Packs	Total Price
Raspberry Pi 4 - 4 GB	4GB-9004	1	$55.00	1	$55.00	$55.00	1	$55.00
Raspberry pi HoE HAT	1188	1	$20.95	1	$20.95	$ 20.95	1	$20.95
Netgear 5-Port GB Eth Sw	GS305	1	$20.10	1	$20.10	$20.10	1	$20.10
T15 Torx Screwdriver	83335A62	1	$9.65	1	$9.65	$9.65	1	$9.65
T8 Torx Screwdriver	83335A69	1	$9.22	1	$9.22	$9.22	1	$9.22
T15 Torx Allen Key	55525A26	1	$3.19	1	$3.19	$3.19	1	$3.19
T15 #8-32, 1/2 L SS Torx Screws	91900A194	25	$11.16	14	$0.45	$6.25	1	$11.16
T8 M2.5 x 0.45 mm, 10mm L Torx Screws	92832A178	50	$8.96	4	$0.18	$0.72	1	$8.96
T8 M2.5 x 0.45 mm, 5mm L Torx Screws	92832A173	50	$11.10	4	$0.22	$0.89	1	$11.10
#8-32 Threaded Insert	90741A210	25	$7.80	17	$0.31	$5.30	1	$7.80
Adhesive Bumper, 1/4OD	95495K463	289	$13.05	10	$0.05	$0.45	1	$13.05

Expandable Locking Grommet	4946A25	100	$12.75	1	$0.13	$0.13	1	$12.75
Screw Down Cable Holder	2024N102	25	$3.50	3	$0.14	$0.42	1	$3.50
Male-Female Hex Standoff M2.5x0.45mm	98952A103	1	$1.58	4	$1.58	$6.32	4	$6.32
#8 Cushioning Washer	93650A110	50	$13.99	2	$0.28	$0.56	1	$13.99

Table 1. Bill of Materials for Version 1 of the Platform Devices.

The initial enclosure, version 1.0, met all the requirements of the project however, the design team is continuing the work of optimizing the design. In version 2.0 the only major design change needed is to move the screw hole locations for the retention ring further down to improve access for the screwdriver to the mount ring. The current location is functional but difficult to reach due to the sheet metal of the stand protruding down from the mounting surface, see Figure x. Moving the holes down would also require a structural change to the retention ring to maintain the secure hold on the stand.

Additionally, version 2.0 will be optimized toward 3D printing material usage. The current housing uses 29.3 cubic inches of build material, 7.1 cubic inches of support material, three days of print time, at a cost of $105 per assembly printed. The build material usage is just over one half of a cartridge. The goal is to reduce the material usage down, at minimum to less than one half of a cartridge. Doing so would lessen the cost per print and print time. Options for material use reduction include, reducing the overall wall thickness, changing part shapes to be more efficient, and identifying smaller connectors and power supply plugs. The color of the enclosure will be switched to one like the color of the rest of the stand.

6. Devices Testing

Phoronix (@Colby)

7. Monitoring Tools

The EC devices represent local resources enabling the rollout of software tools supporting additional functionality or enhancing existing functionality. In the case of our project, the primary goal was to enable the management of the EC devices themselves, to enable active monitoring of the infrastructure and the WebEx service at each location and to provide support staff with the capability to perform basic level of troubleshooting without requiring special permissions.

The following requirements were defined for the management and monitoring service:

- **Easy to Manage** – The EC devices should be as much plug-and-play as possible, the data collection transparent to the user and the data easy to visualize and interpret. Simplicity is critical in making the solution helpful rather than a burden to the community
- **Pro-Active** – From a monitoring perspective, the necessary data could have been collected by either actively testing resources or by listening to active traffic. Both options have their own merits and in some cases a combination of the two leads to complete solutions. For simplicity purposes and to avoid compliance concerns, it was decided to go with a pro-active approach where the EC devices generate traffic which helps collect the necessary telemetry
- **Light-weight** – The solution should not add relevant burden to the existing infrastructure by requiring security policy changes or by consuming significant infrastructure resources such as bandwidth

- **Secure** – The control and data plane of the tools should be secured, and the data collected, while not facing privacy concerns due to the synthetic nature of the traffic used for measurements, should be managed to the standards education institutions.

For this project, an agent based, SaaS (Software as a Service) tool is used. The SaaS option avoids the need to install and maintain dedicated infrastructure for the management tools. The agent is installed on the Raspberry PI during staging (before the WebEx board is deployed) with all the relevant layer 2 authentication information. When the EC device is powered up, the agent communicates with the cloud-based controller over port 443 and self-registers. The agent-controller communication is secure. Once registered, the agent is ready for scheduled or on demand tests configured via a centralized dashboard (see Figure x).

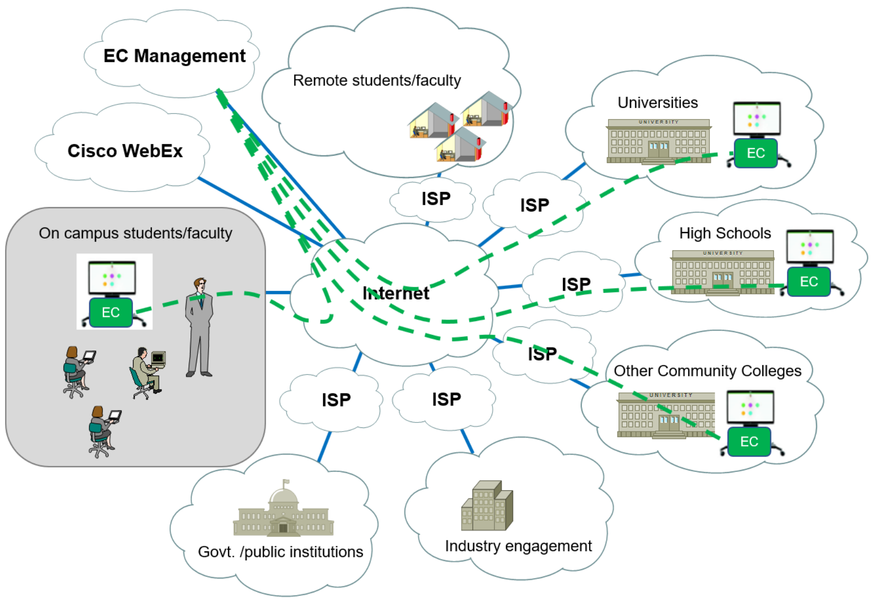

Figure x. EC Platform Management and Monitoring Service Operation

The agents are self-upgrading and they collect data even when they lose contact with the controller. Data collected while disconnected is locally stored and uploaded when the agent automatically re-connects to the controller once it becomes available again.

The agents collect the following information:

- **EC Device Provisioning** – Local IPv4 and/or IPv6 address assigned, local IP gateway. This information helps local staff and stakeholders identify the location of the WebEx board
- **EC Device Health** – Local resources such as CPU, Mem, Disk are monitored to ensure the health of the device and its readiness to collect accurate telemetry
- **Network Path Quality Data** – Path quality information (latency, jitter, drops) is collected using PING [] tests. PING remains an easy, lightweight yet effective way to collect first order network information.

Data is centrally collected and made available to all stakeholders for pre-session preparation, post-session analysis, resource planning and troubleshooting. Examples of the data collected by the monitoring tool used for the proof of concept of the project are shown in Figure x.

<images>

Figure x. Example 1

<images>

Figure x. Example 2

Additional data types are of interest, particularly WiFi quality monitoring and User Experience (UX) monitoring. These data types can help better understand the quality of the infrastructure at that location and enable more in-depth troubleshooting.

Analytics capabilities can be applied to data collected and the project team is particularly interested in offering location assessment options based on historical data and data collected across the entire footprint of the service.

The control and monitoring tool used in the solution development process is helpful in proving the concept and refining requirements. The EC devices however have the resources to support a wide variety of management tools and the project team continues to evaluate options. The results of these investigations are documented in other publications and will be the subject of future work as well.

8. Student Learning Outcomes

Requirements collection and documentation

Collaborative project between students with diverse concentrations

Design strategies

Iterative design

Embedded platform configuration and baselining

Testing tools and processes

9. Conclusions and future work

TBW.

References

1. place holder

2. place holder

3. place holder