Military Applications of Internet of Things: Operational Concerns Explored in Context of a Prototype Wearable

Rune Langleite*†, Carsten Griwodz†, and Frank T. Johnsen*
*Norwegian Defence Research Establishment (FFI), Kjeller, Norway
†University of Oslo, Norway

***Abstract*—Internet of Things (IoT), due to its inherently automated behavior and low development costs, coupled with the emergence of wireless technologies combined with small-sized hardware, has become one of the defining technologies of the last decade. IoT has therefore gained the attention of innovators of military technology, where its role could also prove to be central in gaining information dominance in the battle space. In this paper, a prototype Military IoT (MIoT) soldier wearable was built using commercially available software and hardware, supported by a private network and information-chain built solely out of free open-source software. The communication uses low-power Long Range Wide Area Network (LoRaWAN) communications independent of existing infrastructure, to showcase the ability to provide military deployments with a self-driven, ad-hoc network of sensors. This work was performed in the context of the NATO research task group IST-176 "Federated Interoperability of Military C2 and IoT Systems". In developing the prototype, we interviewed serving military personnel in two rounds: First, to gain important insights on leadership approaches to various military missions, which aided the development of the prototype. Second, to collect feedback on the prototype to conclude whether or not such a system would help increase operational effectiveness.**

The findings show that increased battlespace awareness is possible through automated data acquisition using MIoT. It is therefore recommended that military organizations partaking in such scenarios further investigate the usage of MIoT approaches, specifically including wearables for automating processes that until now constitute fully or semi-manual processes.

I. Introduction

In the past couple of decades, we have seen a surge in ground-breaking, disruptive, and innovative paradigms that changed the way we think of machines and interconnected things. One of them, the Internet of Things (IoT), is quickly gaining footholds in numerous areas. The definition of IoT, as stated by the Global Standards Initiative (GSI) on Internet of Things and International Telecom Union (ITU) standard [1], [2]:

> A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies.

IoT as a business asset has already proven its potential, with an estimated 21.5 billion IoT-specific devices connected to the Internet [3], and an estimated revenue of $11.1 trillion per year by 2025 [4]. Contributing factors to this rise in popularity are, most notably, low development costs and ease of connectivity.

Due to the increased commercial usage of IoT-related technologies, it has become a field of interest for military applications. This interest stems from the importance of information in an increasingly complex and modernized battle space, as stated by the NATO Science & Technology Organization (STO) in its science and technology trends report for the 2020 to 2040 time frame:

> The information domain or info-sphere, is a unique operational environment. This domain is driven by the digitisation and virtualisation of individuals, organisations and societies. [...] 5G and the internet-of-things (IoT) will also increasingly enable the use of the info-sphere.

The report outlines Emerging and Disruptive Technologies, which are anticipated to play a crucial role towards increased operational and organisational effectiveness through, among others, knowledge and decision advantage [5]. In this context, we investigate the usage of IoT through commercially available technologies, in order to establish its applicability within the military. In NATO, multi-national IoT research has been performed in two research task groups: First, IST-147 investigated "Military applications of IoT" (this was also the title of the group). Second, the currently active IST-176 "Federated Interoperability of Military C2 and IoT Systems" is the follow-on to IST-147. The work in this paper has been performed in context of that group.

Currently, military operations are *largely relying on voice communications for effective coordination* between units on the ground.[1] In the heat of battle, information conveyed using voice transmissions often includes mistakes or contains information gaps. This extends to administrative tasks, logistics, medical evacuations, standard reporting, and more. Thus, information dissemination can advantageously be automated further in order to decrease the time spent on voice communications that does not directly relate to combat operations,

[1]It should be noted that this, and the following observations related to use of voice communications, are founded on the operational experiences of the principal author, from his time as military personnel. Further, these considerations of use of voice communication were also confirmed through the interviews we conducted with military personnel as part of the data gathering and analysis for the work in this paper.

such as grids, inventory, etc., which has certain clear benefits. First, it provides combat units with more ability to coordinate their maneuver, rather than spending a lot of time conducting for instance resupplies or providing information to medical units for evacuation purposes. Second, it relieves personnel from manual tasks that traditionally involve heavy human interaction, such as inventory checks and subsequent status updates. Third, it provides a more timely and precise information dissemination, assuming a low presence of false positives and negatives. This can also be combined with Big Data analysis in order to predict when certain needs arise in the future. For instance, given a pattern in resource usage such as fuel and ammunition consumption, automated alerts and tasking can be conducted on behalf of the commanding elements in order to save precious time for the troops in combat. As sensors could theoretically be mounted on any given combat platform, e.g., tanks, war vessels, fighter aircraft, and even individual soldiers, there is a potential for a drastic increase in raw data input to smart systems that could be used for decision making in the battle space, which ultimately should improve combat effectiveness.

In this paper, we focus our work on the rifleman platform through investigating the usability of soldier wearables, which should be used to enhance or augment Situational Awareness (SA) by developing a prototype using Commercial off-the-shelf (COTS) hardware and open source solutions. A wearable can be defined as follows [6]:

> Wearable technology, also known as "wearables", is a category of electronic devices that can be worn as accessories, embedded in clothing, implanted in the user's body, or even tattooed on the skin.

Note that in our work, we focused solely on the least intrusive approach of wearables, i.e., "devices that can be worn as accessories". SA can be defined in very simple terms as an appropriate awareness of a situation, i.e., *knowing what is going on around us*, as M.R. Endsley summarized it [7].

SA is widely considered a crucial foundation for successful decision making in many fields, in particular ones where human safety is of high importance, such as air traffic control, law enforcement, emergency management, and military operations. In this paper, we only consider SA in the context of military operations. Our aim is to investigate civilian IoT approaches in the context of military operations, which is one approach to Military IoT (MIoT). Another approach would be developing military-specific IoT, but this would drive costs up and we would lose one of the major selling points of IoT — that of the low cost.

This paper investigates the applicability of a MIoT subsystem taking the form of a soldier wearable, based on the primary goal of using IoT to improve combat effectiveness through enhanced SA. The research in this paper pursued the following research questions in the context of the three specific cases listed in Section III, not all possible cases that can arise in operations involving the Norwegian Armed Forces:

- R1: How can an IoT wearable improve the current Modus Operandi (MO) in the Norwegian Armed Forces?
- R2: In what way can an IoT wearable enable autonomous information acquisition and dissemination?
- R3: What constitutes a viable approach to a wearable prototype, when emphasis is on low cost, ease of availability and using available civilian technologies?

The remainder of the paper is organized as follows: Related work is presented in Section II. In Section III we present the methodology used, along with the specific cases that frame our experiment. Sections IV and V cover our prototype wearable design and implementation, respectively. The evaluation is discussed in Section VI. Section VII summarizes the main findings, providing answers to our research questions. Finally, Section VIII presents open issues and suggestions for further work.

II. Related work

In this section, we summarize important related work used to frame and limit the scope of our prototype to a viable path using proven technologies for central components.

Standards for generic IoT architectures with detailed specifications are currently an active research topic being conducted by a large group of standardization organizations worldwide [9]. One such example is the IEEE Standard for an Architectural Framework for IoT, which conforms to the ISO/IEC/IEEE 42010:2011 standard for systems and software engineering with respect to architecture description [10]. In terms of military applications and C4ISR (Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance), a framework model nicknamed *IoTNetWar architectural framework* has been proposed, which describes a MIoT system as a four-layered architecture [8] (from the bottom up):

1) Physical Sensing Layer
2) Gateway Communication Layer
3) C4ISR Management Layer
4) Application Layer

The IoTNetWar architectural framework, including prospected technologies towards realization, is visualized in Figure 1. We chose to make use of the IoTNetWar reference model in our work, since it was elaborate enough to cover the architectural components needed to describe and implement a prototype MIoT wearable.

A comparison of different publish/subscribe protocols, i.e., Web Services Notification (WSN), Message Queuing Telemetry Transport (MQTT) and Advanced Message Queuing Protocol (AMQP), showed MQTT to be a very lightweight alternative to the other two protocols when applied in the tactical network [11]. A later study has also shown the feasibility of using MQTT as a protocol in soldier systems on the tactical level [12]. More recently, the NATO IST-150 group titled "NATO Core Services profiling for Hybrid Tactical Networks" [13] has performed extensive experiments with MQTT – both evaluating its performance in emulated tactical networks [14], and also for federated, multi-broker setups [15].

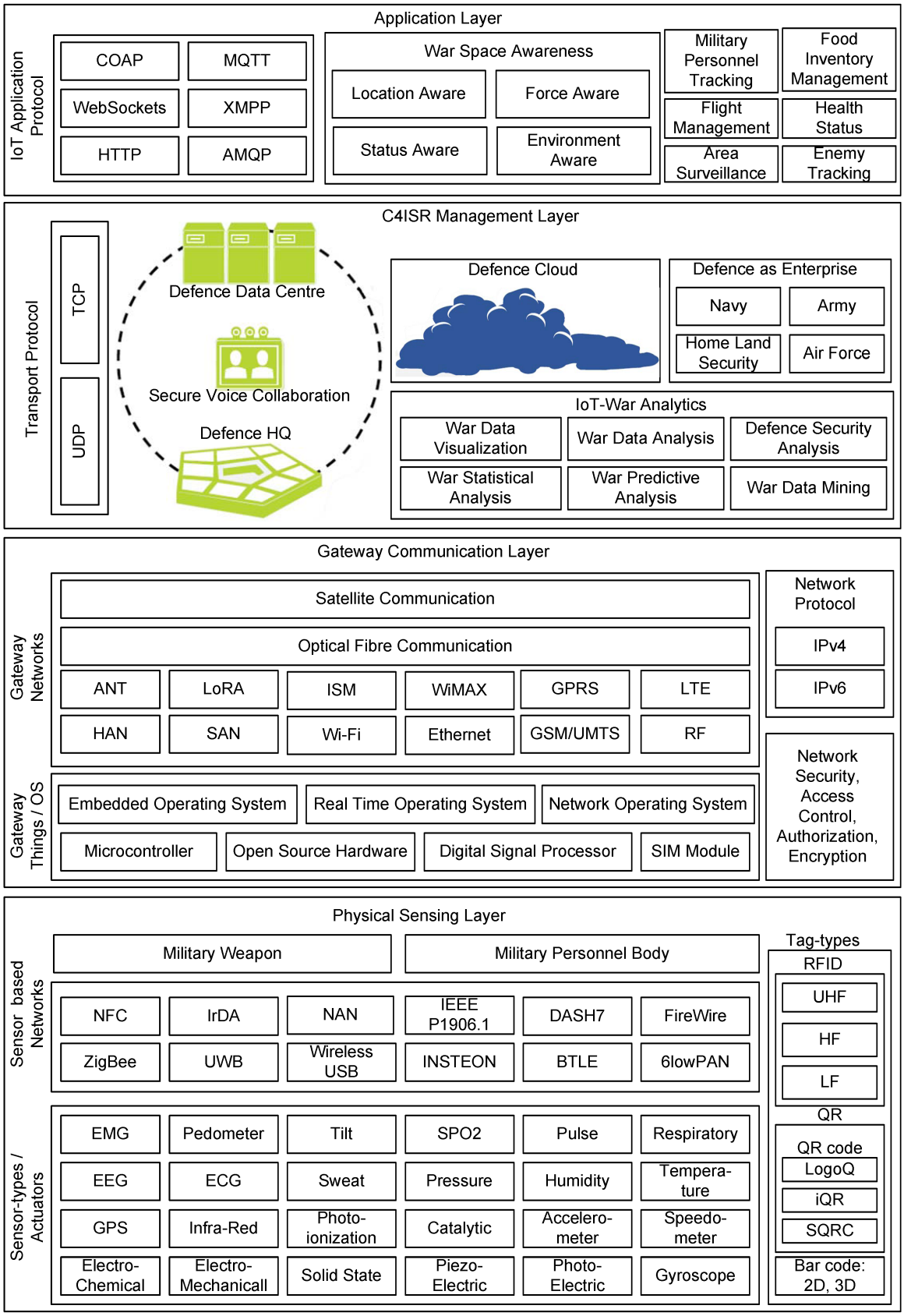

Fig. 1: IoTNetWar architectural framework [8]

So, it is evident that MQTT is a versatile protocol. Furthermore, it is much used for IoT applications, and has been shown to work very well for this purpose by the NATO IST-147 group titled "Military applications of IoT" [16]. Due to these considerations, MQTT was chosen as the publish/subscribe protocol to use for our prototype.

Jalain et al. [17] investigated Long Range Wide Area Network (LoRaWAN) as a protocol in the tactical domain. These experiments used the 915 MHz ISM[2] band. The work had a twofold contribution; it showed the feasibility of using LoRaWAN at the tactical edge, as well as integrating information from the IoT devices with military C2 systems. In this experiment, ranges up to 6.1 miles (approximately 9.8 km) were achieved.

Michaelis et al. [18] evaluated LoRaWAN in an urban environment, for the tracking of vehicles. Like the previous work, the 915 MHz ISM band was used. In these experiments, messages could be received as far as 5.5 km from the gateway. In the case where buildings obstructed the line of sight, packet loss increased and the effective range was shorter, around 2.5 km. Our own experiments in Norway using LoRaWAN in the EU 868 MHz ISM band support these findings [19]. Further, a security evaluation of LoRaWAN that we also performed, shows that the residual risk when using LoRaWAN is acceptable for the specific types of missions we are considering deploying it to [20]. Furthermore, operating in the ISM band, LoRaWAN components can be freely deployed and used, without being reliant on existing infrastructure. Due to these aspects, we consider LoRaWAN an appropriate MIoT communications approach, and leverage it for our prototype in this paper.

III. Methodology

The development process used a hybrid methodology using both software engineering principles and a qualitative approach based on semi-structured interviews. The software engineering methodology is described in [21], outlining distinct steps for designing and implementing an IoT system. Notably, the first set of steps leads to a system design, which then later is implemented.

To aid developing and evaluating developing the prototype, we interviewed serving military personnel in two rounds: The first round was done in the design phase of the project, where we obtained important insights on leadership approaches through the interviews. Later, we implemented the design and evaluated it from a technical viewpoint. Then, the second round of interviews provided subjective feedback on the prototype, helping us to conclude whether or not such a system would improve operational effectiveness.

We considered three specific use cases for the wearable, to limit our focus and also to get a well-defined discussion framework for the interviews. Specifically, we considered using an MIoT wearable for

1) Social patrol in urban environment
2) Urban assault against a fortified enemy
3) Long Range Recon Patrol (LRRP)

These cases were chosen based on their differing nature, where social patrols are meant to establish a point of reference in terms of information flow and detail in situations where there is no immediate hostile activity. Then, in one extreme, the urban assault case implies a high-intensity operation involving a large number of personnel conducting a complex and high-risk task. Conversely, at the other extreme, LRRP is a slow-paced operation involving very limited personnel, where avoiding detection by enemy forces is imperative.

In light of the obviously differing natures between these cases, we investigate the applicability of a MIoT subsystem such as the soldier wearable for the purpose of enhanced SA.

The complete details on these cases, as well as the interview guide, along with transcribed interviews, can be found in [22].

IV. Design

The wearable was designed to attempt automated data acquisition on certain rifleman data that is considered important for commanding elements, or otherwise requires a significant amount of time spent on voice communications to keep commanding elements up to date. In this case, geographical positions and biometrics were considered to be some of the most important data that commanding elements would take an interest in. Other fields such as ammunition-, battery-, and water levels were considered as well, but due to time limitations, we limited the sensor input to a small subset in order to quickly realize a proof of concept of the soldier wearable.

A. Pre-established technology choices

The systems design was developed within the technical constraints already mentioned, e.g., part of the system should use the following IoT baseline:

- **Information exchange:** Java Script Object Notation (JSON) [23], a human-readable, easy to parse and generate, lightweight data-interchange format.
- **Dissemination protocol:** MQTT
- **Waveforms:** WiFi and LoRa (i.e., using LoRaWAN)

B. Interviews

As part of establishing the system design, two semi-structured interviews were conducted for the purpose of fact finding prior to system design, and evaluation of the developed prototype. Both interviews involved three serving military officers (denoted as INF1 through 3) of different, relevant operational backgrounds. In the fact-finding interview, a preliminary design for a soldier wearable was presented as a starting point for the informants to showcase its potential value. In the design, position data, biometrics, and an ammunition counter was used, as it was found that individual soldiers whereabouts and their combat effectiveness was deemed most important.

[2]The term Industrial, Scientific and Medical (ISM) generally refers to equipment or devices that utilize radio frequency energy to perform work. The ISM radio bands are frequencies reserved internationally for the use of radio frequency (RF) for other purposes other than telecommunications, and are limited to certain frequency bands.

The three use cases previously mentioned were used to establish a common foundation in the discussions involving the three informants, whereas in the evaluation interview we used a simulated group of nodes imitating the prototype behavior in order to showcase the wearable to the informants, thus acquiring concrete feedback regarding its applicability. In the simulation, a small-scale, pre-programmed scenario is played out, where a foot-mobile infantry patrol of five members is moving in formation through the terrain, before indicating that the patrol was caught in enemy contact, resulting in one wounded soldier. The scenario finishes with the patrol conducting tactical withdrawal from the enemy contact point, before finally forming a defensive holding position some distance away.

The findings from the fact-finding interview include both technical aspects and cultural concerns. From analyzing the interviews, we found the following main take-away points:

1) The level of detail presented by the high-level design idea gained positive responses from the informants. Thus, the same level of detail should be implemented in the prototype to determine whether or not it is of operational value. However, raw biometric data may not provide commanders with improved SA, as this could lead to information overload, where the data would need to be interpreted in context of the situation to that of the wearer.
2) The provided data must be filtered and aggregated at an appropriate level in accordance to the viewing audience. For this paper, we are considering officers at GFC or OPSOFF positions as these were used in the interview cases.
3) Hostile EW has been identified as the prime counter-argument against implementing autonomous sensing across the whole military organization. At the level where the soldier wearable resides, local EMCON (i.e., radio silence) should be in place in situations where it is necessary to attempt to avoid detection by RF emissions.

These items are considered to be core requirements for the wearable prototype. This was leveraged in the further design of the prototype soldier wearable.

C. Wearable overall design

The high-level architecture of the proposed solution can be seen in Figure 2, where at the far left, the sensors we want to integrate and test are highlighted in red, and others that were considered but not implemented are highlighted in blue. In the design phase following the interviews, it was found that implementing a shot counter for keeping tabs on the ammunition status would be rather difficult due to the lack of equipment and an approved shooting range. Thus, the shot counter was substituted with a gas detection sensor instead, as a means to showcase automatic detection of CBRN (Chemical, Biological, Radiological, and Nuclear) threats. Inspired by suggested solutions outlined in related work [16], [24], [25], the sensor kit included a GPS, biometric sensors for ECG[3] and EMG[4], as well as a sensor for gas detection. As the prototype was developed using commercial, civilian IoT equipment, the gas detector is not designed to detect military-grade weaponized gases. Rather, it is able to detect the presence of gases aimed at industrial- or work environments for health and safety purposes, such as carbon monoxide and ethanol.

D. Operational view specification

The operational view was taken from the software engineering methodology used in this paper. It is a logical model that describes concrete options for an operational implementation. Examples of concrete operations include service hosting, storage, devices, applications and so forth. These are based on a functional view, which serves as a way to logically group functionalities (i.e., FGs — Functional Groups) with instances of concepts defined in the domain model. The FGs, as defined in [21], can be described as follows:

- **Device:** Contains devices for monitoring and control. For the MIoT subsystem, the devices are the end-nodes with mounted sensors.
- **Communication:** Handles the communication for the IoT system, including the protocols that form the backbone of the IoT system and enable network connectivity. In the MIoT subsystem, this is, in order starting with the end-node, the LoRa radio link, WebSockets over WiFi for the backhaul link, and MQTT for the remaining components.
- **Services:** Includes various services involved in the IoT system such as services for device monitoring, device control services, data publishing services, and services for device discovery. The LoRa link requires a service interface on the device that handles the message preparation and transmission, and the gateway will need a similar interface, which forwards these received messages to an intended LoRaWAN Network Server (LNS).
- **Management:** Includes all functionalities that are needed to configure and manage the IoT system. In particular, the device and gateway will need to be able to be remotely controlled and reconfigured on-demand.
- **Security:** Includes security mechanisms for the IoT system. In this case, the built-in security mechanisms of LoRaWAN and the other COTS products we are using.
- **Application:** Includes applications that provide an interface to the users to control and monitor various aspects of the IoT system.

As specified by the software engineering methodology, we can select concrete technologies pertaining to the various FGs as a final step prior to the actual implementation phase, as shown in Figure 3. This mapping structure shows technologies that adhere to the previously described baseline, in addition to prototype-specific elements. At the edge, we specify Arm Mbed OS, which functions as the integration component, i.e.,

[3]ECG is a technique for evaluating heart activity through the electrical activity of the heart muscles.

[4]EMG is a technique for evaluating muscle activity through the electrical activity in skeletal muscles.

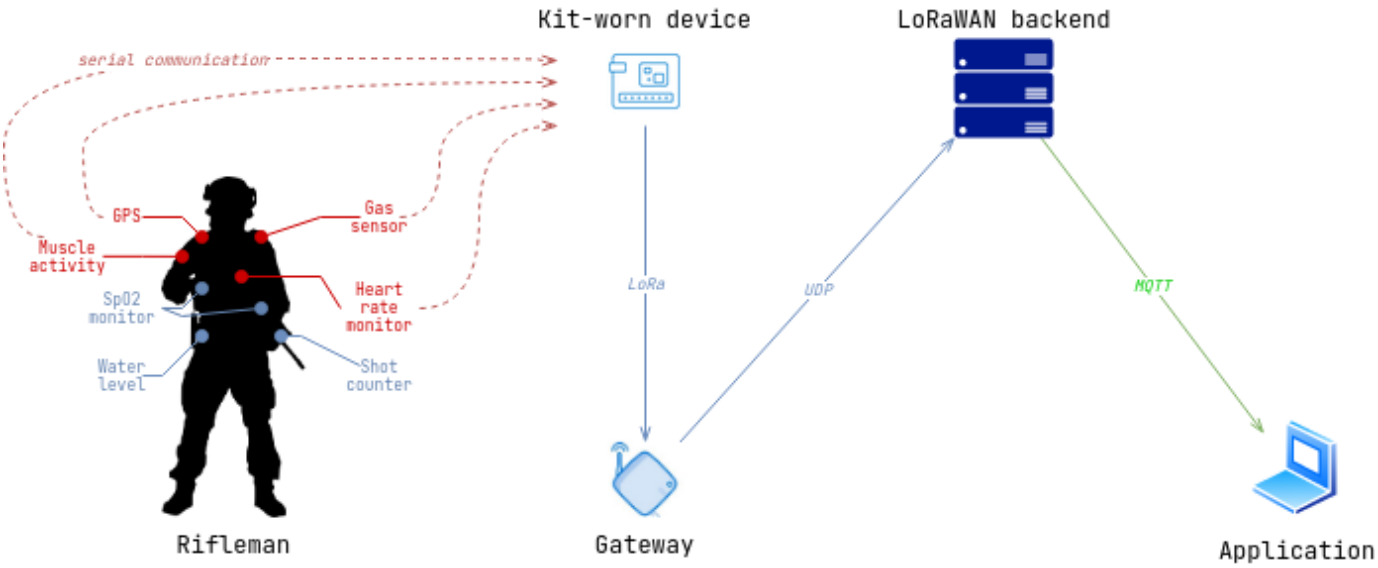

Fig. 2: Soldier wearable high-level architecture

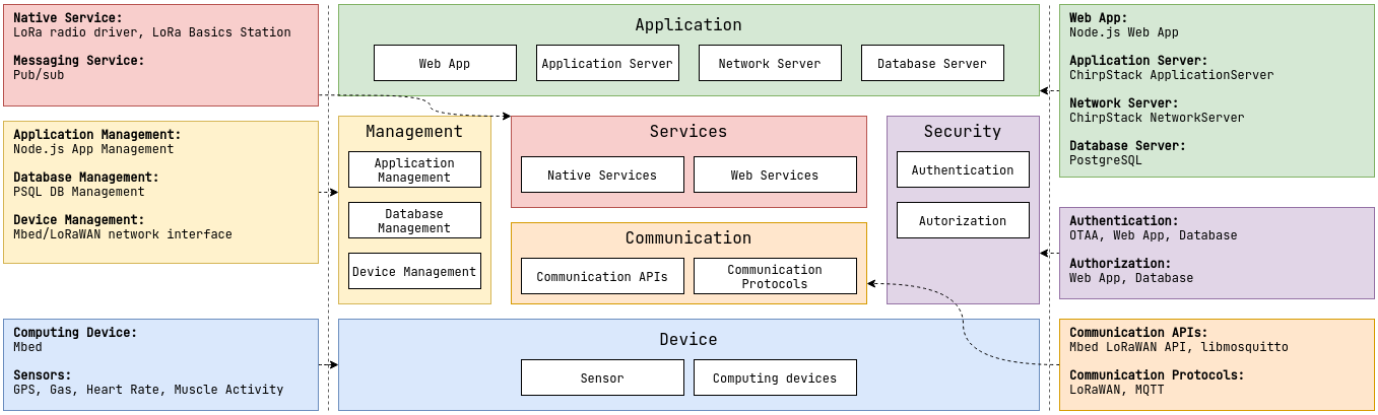

Fig. 3: Operational View

the device that handles sensor readings and transmits them over the air using LoRaWAN. At the cloud segment, we specify ChirpStack and a custom data integration component to support application-specific use cases, which serves as the LoRaWAN backend and data filtering, respectively. Finally, we specify a Node.js application at the user level, which mimics basis functions of a real-world Battle Management System (BMS).

V. Implementation

In this Section, we present our prototype implementation, which is described in the frame of the four distinct layers of the IoTNetWar reference model (recall the four layers, as shown in Figure 1 — we cover these bottom-up in the discussion below).

A. Physical Sensing Layer

The physical layer outlines a number of sensors and actuators, classified under the labels "weapon" or "personnel body", of which only the latter applies in this particular subsystem, using GPS, ECG, and EMG. The local network (i.e., between the integration component and the devices) uses serial communication, namely UART and I2C, and analog signals (i.e., voltage level readings).

An Arm Mbed OS enabled DISCO-L072CZ-LRWAN1 development board was used as the integration platform. This board has a wide range of header connectors, including connector support for the Arduino Uno Revision 3 form factor. A complete overview of the board can be seen in the Mbed OS board overview [26].

The main thread on the device is based on the official Mbed LoRaWAN example implementation [27], slightly modified to handle message transmission and reception in accordance to our use case. Before it can be successfully initialized, a configuration file, which specifies LoRa radio module settings and LoRaWAN settings, needs to be specified [28].

For debugging purposes, it was found useful to enable the full standard printf library, which as of Mbed OS version 6.0 is disabled by default to limit ROM usage [29]. In addition,

to enable CPU statistics (i.e., sleep, deep sleep, and active time metrics), the Mbed OS configuration file needs to be modified accordingly. By default, the Sleep Manager API puts the device to sleep when no threads are active, where either sleep or deep sleep are activated based on a number of criteria [30].

To enable geographic position reporting, an Adafruit Ultimate GPS Breakout v3 [31] was chosen due to its platform support through existing community-developed libraries and its feature-rich capabilities. This GPS module embeds a built-in 64K logger and features command-receptive functionality allowing to tweak its behavior, and outputs standard NMEA 0183 sentences [32] containing location, speed, and altitude data. To communicate with the development board, it requires one UART interface using a fixed baud rate at 9600.

To enable software-controlled information flow, a modified version of the SerialGPS library [33] was used. The library uses the deprecated Serial API to communicate over UART, which should be replaced with an instance of the BufferedSerial in order to utilize software buffers to send and receive bytes to and from the GPS module.

The raw output from the GPS is a continuous data stream of all NMEA formats. Since these sentences are of varying length, we cannot use a fixed-size buffer to acquire the values. However, each sentence starts with a $ character and terminates with a newline character, which could be used to extract NMEA sentences. Once a sentence has been extracted, it is subsequently matched against the GPRMC format, which carries a minimal data set for position information. If the sentence contains valid position data, we chose to convert the latitude-longitude pairs from DMS (Decimal-Minutes-Seconds) to DD (Decimal-Degrees) as we found this format easier to handle. The last step uses the Haversine formula to determine whether or not the newly acquired position deviates from the previous position by at least two meters, which would determine whether or not the position data should be scheduled for transmission.

The Sparkfun AD8233 Heart Rate sensor [34] was found to be the most promising candidate to detect heart beats, largely due to its convenient cable integration for sensor pad placements on the body. As stated in the specification, the HR sensor should be used to calculate a simple BPM value by using ECG, which was done by interpreting the returned analog signal measured in volts from the sensor. A similar implementation using an optical photo-resistor for BPM calculation [35] was used with some slight modifications. The implementation uses a LowPowerTimer instance, which would either run for a maximum of ten seconds or until a heart rate was found (i.e., minimum five beats were successfully detected).

The MyoWare Muscle Activity sensor [36] developed by Advancer Technologies was found to be the best option towards recording muscle activity, which outputs an analog signal representing the rectified and integrated signal of the activity of one single muscle. As this sensor also outputs an analog voltage, a LowPowerTimer instance is utilized to average the muscle activity over a period of ten seconds. This is, however, not a clinically correct MUAP measurement as this would require more sophisticated algorithms, in addition to the fact that we are not continuously measuring the muscle activity. Thus, we implemented a simple averaging measurement to simulate this behavior.

The Grove Multichannel Gas Sensor V2 [37] was used as the gas detector sensor, which communicates over I2C and require a 3.3V power supply, and qualitatively detects a variety of gases through its four on-board gas detection modules:

- GM102B: NO2 (Nitrogen Dioxide)
- GM302B: C2H5CH (Ethanol)
- GM502B: VOC (Volatile Organic Compounds)
- GM702B: CO (Carbon Monoxide)

The provided library for this particular sensor is built for Arduino [38], thus requiring to be ported to Mbed OS for compatibility. This was solved in large part by changing the Arduino-specific TwoWire and SoftwareWire interface libraries with the Mbed OS I2C API [39].

B. Gateway Communication Layer

The gateway communication layer is the link between the physical sensors and the data processing layer, here named the C4ISR Management Layer, which commonly resides in the cloud. We chose a Raspberry Pi 3B running Raspberry Pi OS (previously known as Raspbian), a Debian-based OS built for Raspberry Pi SBCs, to work as the gateway computing platform. An IMST iC880A LoRa concentrator [40] and a SMA antenna with 2 dBi gain was mounted on the Raspberry Pi host, using a LinkLab LoRa gateway shield [41] for convenience.

To enable the LoRaWAN protocol to operate on this particular platform, we chose to use LoRa Basics Station [42] using the concentrator v1.5 station reference configuration, which matches our hardware setup. The local configuration settings need to specify settings for the on-board radio chip, the gateway DevEUI, logging, and preferably a reference to a reset script that resets the concentrator to a clean state, as it is a known bug that the LoRa packet forwarder embedded in the software may become unable to start following hardware restarts or reboots.

Once compiled and configured, the LNS protocol only needs a WebSocket endpoint to connect to the controlling LNS, which later configures the gateway in accordance with the desired LoRaWAN parameters.

C. C4ISR Management Layer

This layer is tasked with specifying the general backend components for a MIoT system, such as data visualization and analysis. However, Big Data and full-fledged data analysis is outside the scope of this project. Thus, a simple application tasked with filtering the data produced by the LoRaWAN Application Server was implemented, effectively acting as a middleware between the LoRaWAN backend and the UI.

The backend infrastructure was set up using a standalone Raspberry Pi 4 installed with the full ChirpStack [43] stack,

namely the Network Server, Application Server, and Gateway Bridge. The Gateway Bridge is ChirpStack-specific, which ensures communication with the gateway over UDP, and subsequently publishes gateway traffic through MQTT using an internal broker.

According to the official ChirpStack documentation, the importance lies with the configuration files to each of the three components to match the local environment. This configuration specifies local database connections, the ISM band, gateway configuration (including channel frequencies), and the join server to be used. At the time of writing, the Application Server acts as the Join server, requiring the JoinEUI on the end-nodes to be set to all 0s.

The Data Integration application, our own implementation realizing basic C4ISR Management Layer functionality, is tasked with filtering the output from the LoRaWAN backend. This component was built as a simple C++ program utilizing the Mosquitto C-library [44] and Nlohmann JSON [45] in order to filter the contents of the data produced by the Application Server. Essentially, it uses a Mosquitto broker to acquire the uplink data, extracts elements that are of interest for the end users and subsequently creates a minimal JSON array, before finally publishing said JSON array on a given topic that the UI subscribes to. In particular, the callsign associated with the wearer is added based on the DevEUI of the device, and a qualitative descriptor is used in place of the received biometric values:

- HEALTHY,
- UNHEALTHY,
- EXHAUSTED, or
- UNDEFINED.

The broker, which the data integration component relies on, also had to be configured to support both plain MQTT and WebSockets simultaneously, as the UI uses WebSockets for its own MQTT connection. Using Mosquitto, this is simply enabled by adding the necessary ports in the local configuration file.

D. Application Layer

The UI was built as a simple web application based on an example Node.js integration by LoRaWAN Academy [46], but extended to use the Eclipse Paho MQTT JS library [47] to receive the filtered uplink data and to schedule downlink commands, in addition to using the Google Maps API [48] to display a map showing sensor location and status. The UI was also implemented to support scheduling of downlink commands and action buttons when certain events arised, such as potentially wounded soldiers as indicated by the biometric descriptor UNHEALTHY.

In this work (as you may recall from our discussion on the design phase and the use of a supporting interview with operational personnel), we are mainly targeting information needs of the first levels of authority in operational settings, i.e., the GFC and OPSOFF. Hence, the UI resembles a BMS as depicted in Figure 4, albeit simplified to only include functions necessary to evaluate the prototype.

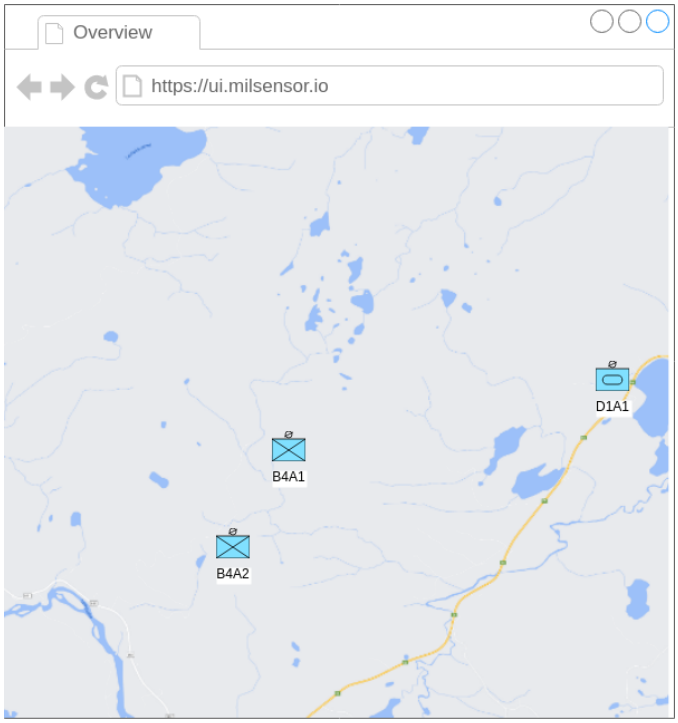

Fig. 4: UI example screenshot

VI. Evaluation

This Section covers both technical assessment and experiences gained from the development process, and assessments based on feedback interviews using the same informants from the fact finding interview. The technical assessment serves as a means towards determining the best suited technologies and approaches, but also providing insights for future versions of soldier wearables.

The subjective evaluation is used to evaluate the informants perceived experience of the system to determine the applicability of the prototype. This was done through the feedback interview described in Section III.

A. Technical evaluation

1) Platform development: Using the official Arm Mbed LoRaWAN example as a starting point, the implementation process was mostly focused on finding proper means to acquire sensor readouts in the context of the LoRaWAN event loop. Initially, manual bit packing of the sensor data was used, but required complex and ineffective decoders at the LoRaWAN backend as the order and size of the data varied. Thus, Cayenne LPP [49] was favoured due to its flexible schema. However, the provided library lacked identifiers for certain data, such as gas readings and biometric values, which therefore needed to use generic identifiers.

2) Sensor integration: The particular hardware platform used in this project worked well for integrating sensors due to its many peripheral connectivity options. Some attention to the pin mappings was however needed as many pins conflicted with each other. For example, the Serial2 TX/RX

pairs conflicted with the STLink connection, and the Serial1 TX/RX pairs conflicted with both I2C and on-board LEDS.

With regards to the gas sensor, unsuccessful attempts were made to detect alcohol fumes. As other gases proved rather hard to acquire and safely test the detection abilities, no further attempts were made. Thus, the usability of the gas sensor remains undetermined.

3) TX logic: The LoRaWAN event loop was implemented in such a way that it would transmit as often as possible, thus providing continuous and timely data updates of the wearer provided that the readout levels passed the required thresholds. If EMCON was active, it would simply remain inactive, but joined to the network, until EMCON was deactivated.

4) Energy conservation: Following each transmission, the CPU usage statistics were collected, which yielded the amount of time the device has spent active and idle since the previous sensor measurements. On average, the device would be idle between 20-30% of the total time period, and thus be in sleep mode. The remaining time period is the active time segment of the CPU. The rather high percentage stems from the fact that all sensor readouts are executed in sequence, and not in parallel.

5) LoRaWAN backend and data integration: ChirpStack was found to be very easy to install, setup, and configure for the platform on which it was running and the system it supported. In particular, the MQTT interface and embedded support for Cayenne LPP made ChirpStack a crucial component for enabling application-wide information flow.

The data integration component also proved relatively simple to implement due to the large community and well-detailed documentation of the frameworks in use. In either direction (i.e., up- or downlink), no MQTT package losses were observed. However, it should be noted that the devices running the LoRaWAN backend and the data integration component both resided on the same local WiFi.

6) User interface application: The UI is largely an ad-hoc application meant only to serve as a visual means towards showcasing the value of the soldier wearable in a BMS-like fashion. As such, no UI/UX considerations were taken.

The UI was implemented with two custom commands for enabling or disabling EMCON on the device. Once activated, the embedded MQTT instance would publish a message to ChirpStack targeting the specific device using the topic *application/1/device/<devEUI>/command/down*. The message was built as a JSON string using a Base64-encoded payload.

In addition, a placeholder button would be displayed if any devices reported unhealthy biometric values, which if clicked displays a message that medical evacuation have been dispatched to the last reported position of the device. The same functionality was implemented for gas detection, where the message displays that CBRN have been dispatched to the same location where this was reported. In practice, this could be implemented in such a way that the message could be addressed to relevant units depending on the situation on the ground. For this paper, however, the functionality was limited to only simulate that such messages were actually sent.

B. Subjective evaluation

1) Prototype feedback: The prototype received positive responses in terms of improving SA on lower levels, thus improving the decision basis for commanders. Due to this, it was suggested that such a system could potentially improve operational tempo, exemplified by fire missions as stated by one informant, where friendly forces locations are known automatically, thus ensuring that artillery strikes do not accidentally hit own forces, as derived from the following statement:

> I can see their position, their formation. The patrol leader can spend more time leading what's happening on the ground rather than keep a report with the rear, because they receive most of the information through this instead. If I as OPSOFF am wondering about something, [...] I can instead look at the screen, where are they, what are they doing, they are doing OK. [...] If something unforeseen happens, then I can prepare resources immediately when something happens, like a QRF or MEDEVAC, [...]. So when I then get voice comms with the patrol leader saying he's in this or that position, then I can press that dispatch button [...] So it's really about increased operational tempo, in addition to increased SA, thus improving the decision basis for the commanders. [...] It greatly improves the tempo on the battlegrounds, so you don't drop artillery on your own forces, you know where not to drive if they are firing in certain directions, and so on. I also think it is useful to be able to zoom in and out to see the units formation and such, since this tells me a lot about their threat assessment. That it updates real-time is also something I appreciate. I also think it is good to be provided with information regarding their state, such as if they are physically exhausted, unhealthy, or healthy, as long as you know what those terms mean.
>
> — INF2

Concerning the UI, it would seem that the informants found the dispatch-feature somewhat disruptive, as it wasn't clear how they would want to use such a functionality. From the responses, it would seem that some voice-based communications would be required regardless if a button-press would, at its core, solve the same task, which in this case is narrowed down to a potential location for evacuation.

2) Suggested operational use: Based on the informants' responses, it is likely that such a system could help close information gaps on lower levels in terms of individual soldiers' whereabouts. It was also suggested that the information provided through such a system might not be as useful in all scenarios, as one informant described traditional full-scale warfare as too intense and too vast in volume for such a system to provide meaningful data in a timely manner. In contrast, one

informant suggested that such a system could be valuable for low-paced missions such as mentoring and/or peacekeeping assignments, as the tolerance threshold for loss of life in these scenarios are far lower compared to full scale war, as derived from the following statement:

> There is a lot that indicates that such a system may produce information overload during high-intensity, steel versus steel, warfare. Where it is a matter of minutes or hours until a unit has either been eliminated or eliminated the enemy. So I think in that case then this might just be an added complexity to the scenario, and not help the SA in any remarkable way. [...] For units conducting stabilization missions or mentoring in for instance Iraq then I think such a system has a completely different role, majorly due to the very low acceptance for loss of life during such international missions compared to the previously mentioned large-scale warfare. So I think it is more in the low-intensity operations that such a system would truly shine, mainly at platoon and company levels.
>
> — INF2

Another informant outlined the information gap such a system could provide by drawing parallels to vehicle tracking systems currently in operational use, as derived from the following statement:

> Something I've really missed as a platoon commander is a live feed of the foot-mobile infantry whenever they were out, where I've had to receive a GPS position from the foot-mobile team leader and plot that manually. So if I as platoon commander have had access to this data in a live feed, then it would have built an incredible SA at platoon, company, and battalion level. It would have been insane amounts of time saved. [...] for contact situations, I think we would have saved, my guess, half an hour.
>
> — INF3

3) Challenges: Based on the responses from both the fact finding and evaluation interview, it is likely that a conservative officer corps could pose some challenges in terms of implementation and active use. Most notably, one informant outlined the possibility that higher military echelons might use such high-resolution information systems to micromanage units on the ground, as given by the following statement:

> I personally know about officers and NCOs that would use this to micromanage them, "go a bit more to the left", "don't go that way", "don't do that", which is a pitfall in itself. But that's more about leadership culture, and not the technology.
>
> — INF2

A clear-cut mitigation for such pitfalls is the level of detail presented to the users based on their position in the military organization. Specifically, the lower an officer is in position in the hierarchy, the higher level of detail they will be able to view, and vice versa, as described by the following statement, talking about information:

> [...] it has to be aggregated. So as a brigade commander, then you see the battalion as a box, and then downwards to the patrol leader that can see all the members of the patrol as individuals. That is absolutely the biggest problem, that leaders get stuck on details they are not really supposed to have. When the brigade commander is interested in what rifleman 1 is doing then he doesn't know his own job.
>
> — INF1

Another notable challenge is the implementation process, where one informant suggested that such a system should be introduced in iterations through small units. For every iteration, the system should be increasingly improved based on the feedback from the users, before finally comparing the unit's performance to other units not utilizing the same system.

VII. Conclusion

Our goal was to investigate the usability of soldier wearables in terms of enhanced or augmented situational awareness by developing a prototype using COTS hardware and open source solutions. In the following, we draw conclusions made through the evaluation phase in the context of the research questions described in Section I.

A. R1: Improving the current MO

This research question was conceptualized in Section IV. The anticipation was that the soldier wearable could help improve combat effectiveness through improved SA and information resolution by visualizing each individual's health status and position. Specifically, the acquired feedback from the subjective evaluation in Section VI highlighted a potentially lowered usage of voice-based communication, which would relieve leaders on the ground to focus more on the task at hand. Additionally, the increased battle space information resolution through the provided position data for individual riflemen proved to be a useful feature for officers in the rear as well, in particular with respect to coordinating external resources such as medical evacuation or fire support, which relies heavily on own units' location. Today, vehicles are already utilizing automatic location reports using their own systems. However, these systems are classified and not easily extendable. At the soldier level, it is therefore a cheap, effective, and quick solution to utilize IoT systems such as the one presented in this paper to close this information gap.

B. R2: Autonomous information acquisition and dissemination

This research question was conceptualized through previously identified mission-critical use cases for IoT in the military domain [24], and finally realized in Sections IV and V through system design and -implementation of a wearable prototype in the frame of the identified use cases in Section III. Using the rifleman platform for information acquisition was found to be useful for three particular information categories, namely: Geographical position data, Biometrics, and Logistics, of which the former two were investigated in this paper. The

logistics category covers rifleman inventory, such as ammunition and water. In the fact finding interview, an ammunition counter was suggested as part of the system design, but not pursued in the prototype.

The position data was shown to be most crucial towards extending existing tracking systems to the rifleman as well as armored vehicles, whereas biometric sensor data was shown to be valuable for the end-user using both quantitative or qualitative data, depending on their position in the military organization. Recall that information needs to reach both vertical and horizontal elements, e.g., the commanding officer needs to know what his units are doing, and support units need to know where the requesting units are located.

C. R3: Viable prototype

This research question was conceptualized through previously conducted experiments involving practical hardware and protocol testing using open standards, open source solutions, and COTS equipment (see Section II). The prototype was designed based on fact finding interviews to support the IoT engineering methodology, and later realized, then evaluated, through both the implementation and evaluation phase of the development process, see Sections V and VI. The implementation phase included studying and obtaining COTS hardware that we previously had no experiences with, and combining these with previously studied, proven approaches to build the wearable. It was found that Mbed OS and its LoRaWAN API stack was fairly simple to use for our custom sensor build. The integration challenge was however found in properly integrating the external sensors, and in particular parsing the returned data. Furthermore, Cayenne LPP was found to be the easiest and most flexible way to pack data into a LoRa-message, rather than using manual bit packing, in particular if the transmitted messages did not adhere to a fixed structure. It was however slightly lacking available data identifiers, which ideally would have offered specific identifiers for every possible biometric attribute.

Furthermore, the gateway setup using Raspberry Pi 3B and iC880A LoRa concentrator, linked together using a LinkLab LoRa gateway shield, proved to be a stable and flexible solution when configured to run LoRa Basics Station.

The evaluation also showed that the specified IoT baseline outlined in Section IV worked well for the prototype developed in this paper, with emphasis on LoRa and LoRaWAN as the carrier for the outer elements in the MIoT network. Furthermore, MQTT was found to work very well for this particular system design, where its low overhead and ease of use made implementation of custom solutions a relatively simple task.

Finally, ChirpStack running on a Raspberry Pi 4 proved to be an effective and stable LoRaWAN backend solution, due to its ease of setup and configuration. The built-in support for Cayenne LPP also proved to be a crucial part towards enabling dynamic data transmissions from the end-node to user application.

VIII. Further work

For future work, we think it is essential to move to the "next level" of wearables, i.e., to sensors embedded into clothing. For the work discussed in this paper, we had loose sensors worn on the body. While it does work, it is cumbersome to attach them and wires may adversely affect movement. So, for a next generation prototype, the sensors need to be integrated into clothing.

Having additional functionality (e.g., logistics) as well as more sensors would also be useful. Also, investigating edge computing approaches, to get a better approach to processing, disseminating and integrating the data flow from sensors to C2 systems. Finally, considering interoperability aspects, the choice of protocols and data formats for information exchange remain an open issue. For this particular prototype we implemented something that could be used nationally, but for future work, this needs to be aligned with use in a federation of systems. In NATO, IST-176 is working on this issue, investigating approaches to Ontology and Domain specification for IoT/C2 Services.

References

[1] Global Standards Initiative (GSI), "Internet of things global standards initiative," https://www.itu.int/en/ITU-T/gsi/iot/Pages/default.aspx, Accessed 2021.01.25, 2015.

[2] ITU, "Overview of the internet of things, ITU recommendation y.4000/y.2060," International Telecommunication Union (ITU), Recommendation, 2012. [Online]. Available: https://www.itu.int/rec/T-REC-Y.2060-201206-I

[3] Statista Research Department, "Global number of connected iot devices 2015-2025," https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/, Accessed 2020.02.02, 3 2020.

[4] J. Manyika, M. Chui, P. Bisson, J. Woetzel, R. Dobbs, J. Bughin, and D. Aharon, "Mapping the value beyond the hype - executive summary," *McKinsey Global Institute - The Internet of Things*, 6 2015.

[5] D. F. Reding and J. Eaton, "Science & technology trends 2020-2040," *NATO Science & Technology Organization, Office of the Chief Scientist, Brussels, Belgium*, 2020. [Online]. Available: https://www.nato.int/nato_static_fl2014/assets/pdf/2020/4/pdf/190422-ST_Tech_Trends_Report_2020-2040.pdf

[6] A. Hayes, "Wearable technology," https://www.investopedia.com/terms/w/wearable-technology.asp, Accessed 2021.04.10, 2020.

[7] M. Endsley, "Theoretical underpinnings of situation awareness: A critical review," Situation awareness analysis and measurement, Jan. 2000, pp. 332.

[8] P. P. Ray, "Towards an internet of things based architectural framework for defence," in *2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*, 2015, pp. 411–416.

[9] Next-Generation Internet of Things (NGIoT), "Standardization bodies," https://www.ngiot.eu/community/standardization-bodies/, Accessed 2020.02.02, 2020.

[10] ISO, "Systems and software engineering - architecture description, ISO/IEC/IEEE 42010:2011," International Organization for Standardization (ISO), Standard, 2011. [Online]. Available: https://www.iso.org/standard/50508.html

[11] IST-118, "SOA Recommendations for Disadvantaged Grids in the Tactical Domain," Final Report of IST-118. DOI 10.14339/STO-TR-IST-118. Published June 2020.

[12] M. Manso, F. T. Johnsen, K. Lund, and K. S. Chan., "Using MQTT to Support Mobile Tactical Force Situational Awareness," IEEE ICMCIS 2018, Warsaw, Poland, 22nd – 23rd May 2018.

[13] N. Jansen, M. Manso, A. Toth, K. S. Chan, T. H. Bloebaum, and F. T. Johnsen, "NATO Core Services profiling for Hybrid Tactical Networks — Results and Recommendations," IEEE ICMCIS 2021, May 2021, Virtual only.

[14] F. T. Johnsen, T. H. Bloebaum, N. Jansen, G. Bovet, M. Manso, A. Toth, and K. S. Chan, "Evaluating Publish/Subscribe Standards for Situational Awareness using Realistic Radio Models and Emulated Testbed," 24th International Command and Control Research and Technology Symposium (ICCRTS), October 29-31 2019, Laurel, Maryland, USA.

[15] F. T. Johnsen, M. Manso, and N. Jansen, "Evaluation of Message Broker approaches for Information Exchange in Disadvantaged Tactical Networks in a Federated Environment," International Command and Control Research and Technology Symposium (ICCRTS), 2020, Virtual only.

[16] F. T. Johnsen, Z. Zieliński, K. Wrona, N. Suri, C. Fuchs, M. Pradhan, J. Furtak, B. Vasilache, V. Pellegrini, M. Dyk, M. Marks, and M. Krzysztoń, "Application of iot in military operations in a smart city," in *2018 International Conference on Military Communications and Information Systems (ICMCIS)*. IEEE, 2018, pp. 1–8. [Online]. Available: http://dx.doi.org/10.1109/ICMCIS.2018.8398690

[17] B. Jalaian, T. Gregory, N. Suri, S. Russell, L. Sadler, and M. Lee, "Evaluating LoRaWAN-based IoT devices for the tactical military environment," 4th IEEE World Forum on Internet of Things (WF-IoT) 2018, Singapore, February 5-8, 2018, 2018.

[18] J. Michaelis, A. Morelli, A. Raglin, D. James, and N. Suri, "Leveraging LoRaWAN to Support IoBT in Urban Environments," IEEE World Forum on Internet of Things (WF-IoT) 2019, 16 April 2019, Special session on military applications of IoT, Limerick, Ireland.

[19] F. T. Johnsen and T. Søndrol, "Asset Tracking Using LoRaWAN: Experiments Concerning Effective Range and Signal Interception," International Command and Control Research and Technology Symposium (ICCRTS), 2020, Virtual only.

[20] F. Mancini and F. T. Johnsen, "A Novel IoBT Security Assessment Framework: LoRaWAN Case Study," International Command and Control Research and Technology Symposium (ICCRTS), 2020, Virtual only.

[21] A. Bahga and M. Vijay, *Internet of things: A Hands-On Approach*. Arsheep Bahga & Vijay Madisetti, 2014, ch. 5, pp. 99–120.

[22] R. Langleite, "Military applications for IoT: Utilizing soldier wearables for enhanced battle space Situational Awareness," Master's thesis, University of Oslo, May 2021.

[23] ECMA, "The json data interchange syntax, ISO/IEC 21778," European Computer Manufacturers Association (ECMA), Standard ECMA-404, 2017. [Online]. Available: https://www.ecma-international.org/publications/standards/Ecma-404.htm

[24] P. Fraga-Lamas, T. M. Fernandez-Carames, M. Suarez-Albela, L. Castedo, and M. Gonzalez-Lopez, "A review on internet of things for defense and public safety," *Sensors*, vol. 16, no. 10, p. 1644, 10 2016. [Online]. Available: http://dx.doi.org/10.3390/s16101644

[25] N. Suri, M. Tortonesi, J. Michaelis, P. Budulas, G. Benincasa, S. Russell, C. Stefanelli, and R. Winkler, "Analyzing the applicability of internet of things to the battlefield environment," in *2016 international conference on military communications and information systems (ICMCIS)*. IEEE, 2016, pp. 1–8. [Online]. Available: http://dx.doi.org/10.1109/ICMCIS.2016.7496574

[26] Arm Mbed, "Disco l072cz lrwan1 development board overview," https://os.mbed.com/platforms/ST-Discovery-LRWAN1/, Accessed 2021.03.30, 2017.

[27] ——, "Example lorawan application for mbed-os," https://github.com/ARMmbed/mbed-os-example-lorawan, Accessed 2020.10.20, Arm, 2017.

[28] ——, "Mbed os lorawan configuration," https://os.mbed.com/docs/mbed-os/v6.9/apis/lorawan-configuration.html, Accessed 2021.03.31, 2021.

[29] H. Kamba-Mpiana, E. Donnaes, B. Szatkowski, and R. Kanagaraj, "Minimal printf and snprintf," https://github.com/ARMmbed/mbed-os/blob/master/platform/source/minimal-printf/README.md, Accessed 2021.04.02, Arm, 2019.

[30] Arm Mbed, "Power optimization," https://os.mbed.com/docs/mbed-os/v6.9/apis/power-optimization.html, Accessed 2021.03.31, 2019.

[31] Adafruit Industries, "Adafruit ultimate gps breakout v3," https://www.adafruit.com/product/746, Accessed 2021.03.25, 2012.

[32] NMEA Standards Committee, "Nmea 0183 interface standard," National Marine Electronics Association (NMEA), NMEA Standard, 2018. [Online]. Available: https://www.nmea.org/content/STANDARDS/NMEA_0183_Standard

[33] Arm Mbed OS Components Team, "Serialgps: Library for em-406 and mtk3339 gps modules," https://os.mbed.com/teams/components/code/SerialGPS/, Accessed 2020.12.20, 2014.

[34] Sparkfun Electronics, "Single lead heart rate monitor - ad8232," https://www.sparkfun.com/products/12650, Accessed 2021.03.25, 2016.

[35] M. A. Ionascu, "Heart rate monitor," https://os.mbed.com/users/maryannionascu/code/HeartRateMonitor/, Accessed 2021.02.23, 2015.

[36] Advancer Technologies, "Myoware muscle sensor datasheet (at-04-001)," http://www.advancertechnologies.com/p/myoware.html, Accessed 2021.02.20, 2015.

[37] Seeed Technology Co. Ltd., "Grove multichannel gas sensor v2," https://wiki.seeedstudio.com/Grove-Multichannel-Gas-Sensor-V2/, Accessed 2021.02.25, 2018.

[38] W. Weng, "Grove multichannel gas sensor v2 arduino library," https://github.com/Seeed-Studio/Seeed_Arduino_MultiGas, Accessed 2021.02.25, Seeed Technology Co. Ltd., 2019.

[39] Arm Mbed, "I2c api reference," https://os.mbed.com/docs/mbed-os/v6.9/apis/i2c.html, Accessed 2021.02.25, 2021.

[40] IMST, "ic880a-spi lora concentrator," https://wireless-solutions.de/products/lora-solutions-by-imst/radio-modules/ic880a-spi/, Accessed 2021.02.08, 2021.

[41] CH2I, "Lora gateway shield," https://github.com/ch2i/iC880A-Raspberry-PI, Acessed 2021.02.02, 2018.

[42] A. Beitler and A. Singh, "Lora basics station," https://doc.sm.tc/station/, Accessed 2021.02.01, Semtech Corporation, 2019.

[43] O. Brocaar, "Chirpstack architecture," https://www.chirpstack.io/project/architecture/, Accessed 2021.04.07, 2019.

[44] Eclipse Foundation, "libmosquitto — mqtt version 3.1.1 client library," https://mosquitto.org/api/files/mosquitto-h.html, Accessed 2021.03.23, 2010.

[45] N. Lohmann, "Json for modern c++," https://json.nlohmann.me/, Accessed 2021.03.20, 2013.

[46] Semtech Corporation, *Building a custom integration*, LoRaWAN Academy, 2019. [Online]. Available: https://lora-developers.semtech.com/learning-center/lorawan-academy/courses/building-a-custom-integration-1

[47] Eclipse Foundation, "Eclipse paho javascript client," https://www.eclipse.org/paho/index.php?page=clients/js/index.php, Accessed 2021.03.23, 2013.

[48] Google, "Google maps javascript api," https://developers.google.com/maps/documentation/javascript/overview, Accessed 2021.03.23, 2009.

[49] myDevices Inc., "Cayenne low-power payload," https://developers.mydevices.com/cayenne/docs/lora, Accessed 2020.12.05, 2018.