2023 24th IEEE International Conference on Mobile Data Management (MDM) | 979-8-3503-4101-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/MDM58254.2023.00061

# Tackling Network Challenges in Context Aware Environments: Lightweight Context Management Architecture

1st Shaine Christmas
*School of Information Technology*
*Deakin University*
Geelong VIC, Australia
schristmas@deakin.edu.au

2nd Robert Davidson
*Defence Science and Technology Group (DSTG)*
Edinburgh SA, Australia
Rob.Davidson@defence.gov.au

3rd Arkady Zaslavsky
*School of Information Technology*
*Deakin University*
Geelong VIC, Australia
arkady.zaslavsky@deakin.edu.au

4th Kevin Lee
*School of Information Technology*
*Deakin University*
Geelong VIC, Australia
kevin.lee@deakin.edu.au

***Abstract*—Resilience in context-aware applications is especially important within challenged network and physical environments. This paper discusses the current methods for maintaining resilience in context management architectures. These methods largely focus on resolving contextual information loss, rather than maintaining the standard operation of the deployed system. The paper contends that this approach leads to loss of functionality and unwanted modifications to contextual information. Existing approaches for mitigating the effects of network instability have a high resource requirement and do not maintain the standard functionality of the system in real time. The paper proposes and discusses the Lightweight Context Management Architecture (LCMA) which addresses the lack of lightweight solutions for resilient context management systems. The functional requirements of the LCMA components are proposed and detailed. The proposed LCMA will be validated in mission-critical applications with potential adversarial actions.**

***Index Terms*—Context, Context Management Architecture, Network Resilience, IoT.**

## I. Introduction

Context and context-aware systems are an increasingly common area of investigation [17]. As implementation experience grows, real-world context system deployment issues not necessarily seen in simulated test and development environments have been identified. For example, as applications become extensive in scope and geography [6], network issues can cause a breakdown of traditional centralised systems. While context relevant issues have been explored previously [18], solutions applied to larger context systems to date often require more resources, particularly computational power [2]. These systems also have reduced functionality as stored context cannot be used while disconnected from the central management system. This can impair the ability of the given system to complete its objectives.

One limitation of context applications is network connectivity [9]. When network stability is jeopardised, context systems cease functioning in their intended manner. Minimising the data corruption and work required for a context management system to access data is paramount to ensuring normal function in the deployed network [8]. The inherent limitations of context applications can be reduced with network and application design resiliency. Taking account of these issues, especially in the context of network resilience, can help to increase the use and application of context-aware systems [16].

This work aims to explore and address the problems in context management platforms concerning network resilience.

An architecture is presented for a dynamic, pseudo-centralised context system that can continue functioning when network stability is compromised. The contributions of this work are: (i) a review of context management and network resilience issues; (ii) motivation for the development of a multi-agent context architecture to solve presented issues; (iii) a proposed Lightweight Context Management Architecture (LCMA) for distributed, multi-agent context management.

The remainder of this paper is as follows: Section II discusses context management and relevant network stability issues. Section III discusses motivations for a resilient context management architecture. Section IV discusses the parts of a multi-agent context architecture and outlines the requirements for an LCMA. Finally, Section V concludes this work.

## II. Context Awareness

Multiple definitions of contextual data exist [15], often to serve their respective work area without referring to usage in the wider industry. A general definition of context is presented: any information or data pertaining to an environment. Depending on the application, collected data may relate to the physical environment [3], digital environment [13] and other unique events within an environment [1].

2375-0324/23/$31.00 ©2023 IEEE
DOI 10.1109/MDM58254.2023.00061

Devices within a context system are split into two classifications [24]: (i) context providers focus on supplying contextual data to the management platform, while (ii) context consumers request contextual data for use.

A Context Management Platform (CMP) focuses on managing contextual information [10], stores context data long term and routes data from context providers to context consumers. Context management platforms can contain context collation, context storage, and decision-making systems. A Context Management Architecture (CMA) links a CMP with context collectors, providers and consumers.

When context networks become more extensive, a single context management device will be unable to meet the system's needs. There are two approaches to this problem. Firstly, distributed context awareness [12] takes the functions of a traditional CMP and distributes those functions across multiple devices, which can differ in service area. If one part of the distributed CMP becomes non-functional (either through power loss or network disruption), the rest of the system can be negatively affected. Secondly, multi-agent context awareness [23] focuses on smaller deployments of CMPs, whereas larger platforms bring together collected context. If a local CMP is unusable, the rest of the system is only affected by the local data loss.

As context systems are deployed, more comprehensive applications may face problems with traditional networks, such as network stability, data loss and corruption over longer distances, and malicious actors if deployed in public spaces [5]. For context systems, the time and stability of collected data are essential to ensure the system can perform in its intended manner [21].

One method of coping with information loss from network instability is context caching. Context caching [11] allows for storing context locally. When applied to network outages, context caching can store data being collected by providers until a centralised CMP can be reached. While this does help to prevent overall data loss, this solution is only suitable for short-term disruptions. For more extended outages or prolonged disruptions to the network, more extensive storage resources would be required by context providers. Context consumers cannot receive context in these circumstances, stalling regular operation until the connection between the providers, consumers, and the central CMP is restored.

Similarly, imputation is a technique used to help mitigate the effects of data loss. Imputation allows for the substitution of lost data [14]. While the substitute data can be generated using machine learning techniques [20], imputation does not use existing context providers - changes in the environment will not be reflected in the substituted data. Context consumers that become disconnected cannot receive information from the remainder of the system and cannot act upon the substituted or collected context.

When a path between a context provider and a CMP is disrupted, and another path can be identified [19] using existing infrastructure, those nodes can act to forward data onward. While this solution restores the connection between the provider and CMP, it requires additional infrastructure to support providers and consumers. It also results in a further delay between data collection and data received by the CMP, which could be critical depending on the time sensitivity of the data.

## III. Motivation

A range of different applications can use contextual information to assist in more rapid and context appropriate decision making. For example, Search and Rescue (SAR) operations could benefit from constant context collection from devices and reactionary decision making of context aware agents. The following scenarios aim to demonstrate the potential application of a context system to an emergency services response and identify current problems in context management due to pitfalls in design and architecture.

### *A. Scenario 1: Semi-Automated Fire Search and Rescue*

SAR operations in hostile environments, such as bushfires or flooding, can place SAR teams in dangerous conditions. This danger can be mitigated by locating the victims before deploying SAR personnel within the high-risk region. A context system could control an automated search system using autonomous all-terrain agents. Collected context can help to identify hazards and paths for emergency workers, discover unknown victims, and pinpoint conditions for victim status and evacuation feasibility.

A traditional centralised context system would struggle to operate in this scenario due to the hostile conditions that can negatively affect the network connectivity of the context system. Smoke and foliage cover can interrupt network connectivity [4] to individual context providers and consumers, as well as to the central control system, disturbing the movement of agents to targets. Not only would a central system never receive pertinent information to the local environment, but agents would not receive commands, delaying the operation until connectivity is restored.

Consider the situation of a group of autonomous vehicles locating individuals trapped in a forest fire (Figure 1). This situation is ideal for applying a context system; in addition to tracking where individuals may be located using environmental clues, it also allows for identifying a safe escape route and monitoring the fire movement. In this situation, network communications may be interrupted by tree cover and smoke, which can impact communication with a base station. As such, autonomous units must be able to make decisions based on local context alone when disconnected from a central CMP.

Figure 2 outlines what may occur if communications are disrupted between the autonomous vehicles and the central CMP. Not only can the vehicles become damaged and need to return to a safe location, but halting the progress of the mission objective forces the situation to degrade. This illustrates the need for stable and sustained progress towards mission objectives, even when autonomous vehicles can no longer communicate with a central CMP. Missing data may additionally cause autonomous vehicles to misunderstand the

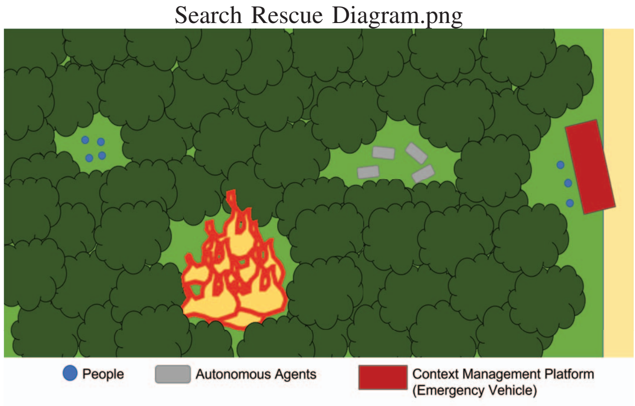

Fig. 1. Fire SAR operations where autonomous context systems would be beneficial.

current state of the local environment. For example, suppose a vehicle receives information from another vehicle that the local area is safe to traverse. A certain level of trust is required to maintain vehicle operation and continue the mission. The overall mission would be affected if this information is false, and the autonomous vehicle could be completely damaged or lost.

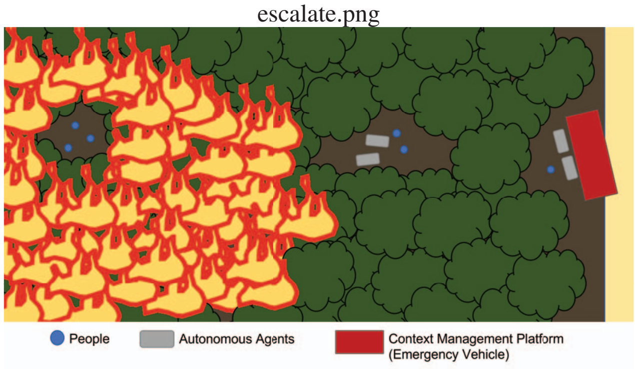

Fig. 2. Fire SAR using autonomous context systems in disrupted environments.

This scenario outlines the need for context systems to remain operational even during network disconnections and outages: to ensure that data collection is timely and relevant and that decisions can still be made during the disrupted operation.

### B. Scenario 2: Surf Life Saving

An automated contextual information detection and analysis system could benefit Surf Life Saving SAR operations. Environmental conditions such as ocean swell can be volatile, impacting the survival time of people lost in that environment and potentially impacting those personnel searching. Context could be used in this situation, with context providers operating on the beach and on the water, sending valuable information to context consumers to assist in the decision making process.

Using a traditional CMA, each context provider can constantly collect information from the surrounding environment. Instability can arise in the connection to the CMP: waves [7], boats, coastal land formations and other obstructions may interfere with communications to the central system. The signal strength can also decrease over longer distances, reducing the geographic spread of possible applications [22]. To avoid total information loss during connection outages, devices must be able to continue collecting contextual information and ensure that when the connection is restored, relevant data is given to the context consumers. The ability of devices to compensate for disconnection by using caching is limited by the context providers' short-term storage and processing ability. The frequency and duration of disruptions to flows dictate the exact requirements.

If we depict the physical layout diagrammatically (Figure 3), it would consist of a CMP for data collection present on the beach, with boats and buoys (stars in Figure 3) on the water able to collect data and act on command from the context-based application. This example represents an idealised situation for using a context management system, as the information is being collected regularly and adds to the existing contextual information of the surrounding environment. When people are in danger, resources can be dispatched on time to facilitate rescue and emergency care. While resources can be human controlled and driven, the allocation of tasks to resources can be determined using the local context. For example, they may be simple calculations of how far away a person is from a rescue craft or more complex calculations to determine how to prioritise people for rescue based on the cost/benefit utilising information such as distance, injury or proximity to danger. When contextual data is collected in this system, other boats and waves can interrupt communications between context providers and the base CMP. This may result in decisions for alerts not being triggered or triggered much later than desired. Other environmental monitoring efforts may be disrupted if collected data is not regular and could affect surveys for dredging and other environmental care efforts.

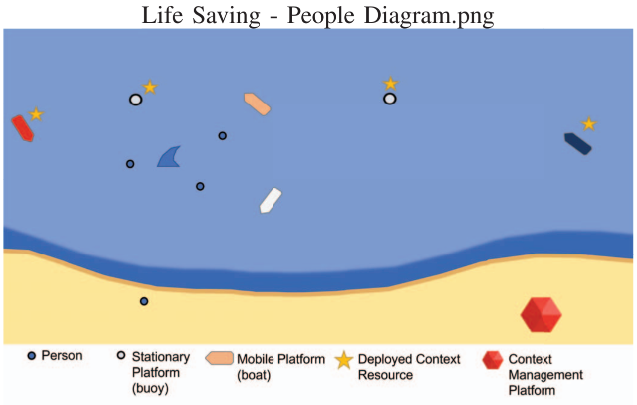
Life Saving - People Diagram.png
Person
Stationary Platform (buoy)
Mobile Platform (boat)
Deployed Context Resource
Context Management Platform

Fig. 3. Ocean monitoring using static and non-static collection resources.

Context providers and the CMP may also be running on different power sources, leaving a point of failure if the CMP cannot operate. An outage for the CMP will result in mass context loss. Any use for collected context (i.e. shark detection, emergency resource allocation, etc.) would cease function until the CMP was restored.

This scenario highlights the need for a context system to continue collecting and managing context during centralised CMP outages. A semi-centralised or decentralised system could help avoid these issues since nodes can operate independently.

### C. Discussion

Each of these scenarios presents an idealised application of context aware systems. Due to the network quality issues in their respective environments, current context aware systems cannot maintain standard operation. To address these shortcomings, a new architecture for context aware systems is proposed to allow operations to continue even when network disruptions occur [16].

Network quality characteristics should be a part of the planning process and influence the creation of connections between nodes in a context aware system, particularly in hazardous and irregular locations. Other solutions to network quality, such as alternate hardwired paths between devices, may increase resilience. These solutions will not work in the listed scenarios due to the irregularity and dynamic changes in asset deployment. Therefore, a solution must be designed so that each deployed agent does not need to rely on network connections to continuously function.

To address these situational challenges, a pseudo-decentralised multi-agent CMA is being developed. This system is designed to operate normally even if the lead node is disconnected or other network disruptions occur. The proposed system needs to address the following identified issues of short and long term disconnections between context providers, consumers and the CMP:

- Informant loss from context providers
- Limitations of cache context data by context providers
- Delay of a decision by context consumer
- No decision by context consumer
- Reduced progress towards system goal as managed by the CMP

## IV. Lightweight Context Management Architecture (LCMA)

To address the issues identified in Section III, an LCMA is proposed. This architecture focuses on deploying context management systems to small scale devices connected directly to context providers and consumers. These context management devices can be networked to a centralised system when a connection is available, continue operating as a standalone unit when connections are unavailable or connect to each other to share context in an ad-hoc decentralised CMP.

The low-level, multi-agent LCMA is outlined in Figure 4. When no network disruptions occur, a central CMP manages all devices. When a device is separated from the remainder of the network, it will continue operating, as it is a self-contained CMP. When disconnection occurs, if the local CMP can connect to other local context management devices, then context sharing can still occur, enabling continued operation.

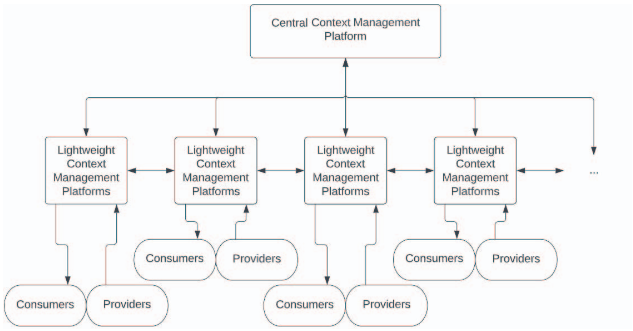
Central Context Management Platform
Lightweight Context Management Platforms
Lightweight Context Management Platforms
Lightweight Context Management Platforms
Lightweight Context Management Platforms
...
Consumers
Providers
Consumers
Providers
Consumers
Providers
Consumers
Providers

Fig. 4. Proposed Multi-agent architecture for LCMA.

A system designed in this manner consists of 2 modules: a heterogeneous lightweight CMP for deployment on low powered devices and a flexible CMP for centralised control.

### A. Flexible Context Management Platform

The flexible CMP operates as a standard CMP when all devices are connected but can discover resources dynamically to allow for device and context system connection/re-connection. As discussed in Section II, a centralised context management system can include context storage, decision making modules, and context collection modules.

In addition to standard context management functions, the flexible platform must allow for dynamic addition and removal of context providers and consumers and backward addition of context when a connection is restored.

To allow for managing the multi-agent LCMA, the flexible CMP will have the following requirements:

- Connect to context providers and consumers
- Receive context, and perform deployment specific functions (context collection and storage, decision making and sending of instructions)
- Map currently available resources, as well as past disconnected resources
- Collate context for storage and semantic conversion.

Several specific modules must exist for a flexible CMP to function (Figure 5). The context discovery module aims to rediscover disconnected devices when outages are resolved. The communication module handles communications between the Lightweight CMPs (LCMPs) and the central system in context delivery and commands. The collation module focuses on collating data from multiple CMPs for further storage. The recovery module integrates data recovered from previously disconnected devices. Each of these modules works together to aid in the overall management of the system. Further modules, such as decision making modules and short term caching, may be added.

Context Management Platform System Architecture - Low Level.png

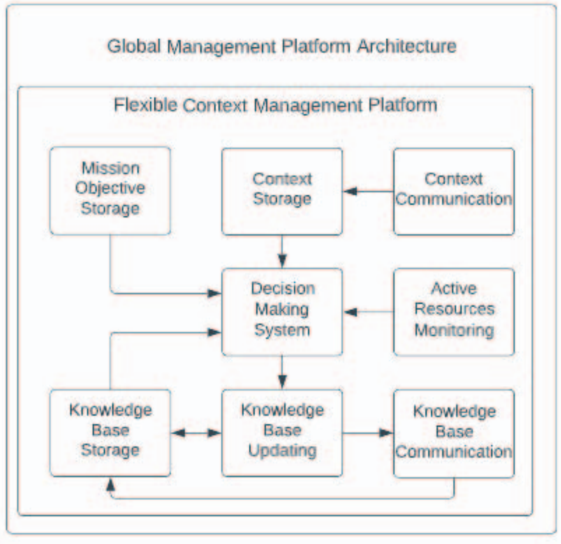
Global Management Platform Architecture
Flexible Context Management Platform
Mission Objective Storage
Context Storage
Context Communication
Decision Making System
Active Resources Monitoring
Knowledge Base Storage
Knowledge Base Updating
Knowledge Base Communication

Fig. 5. Central CMP design architecture.

### B. Lightweight Context Management Platform

To allow for decentralised context management and continued operation during network instability, the LCMP will be deployed on individual context producer and consumer systems. This would enable autonomous agents (as discussed in Section III-A) to operate as a micro CMP, independently of the remainder of the system. During normal operation, this LCMP would work as a collection of context providers and consumers, sending context to the wider CMP and receiving instructions from the wider CMP.

When communication breaks down between the LCMP and the wider CMP, the LCMP must be able to continue to work towards the broader goal of the system. Depending on the application, this may involve context caching for collection based systems and deciding device actuation. If communication can be restored, the LCMP must be able to communicate any stored context and new context to the central system to restore normal operation with minimal disruption.

As discussed in Section III-A, the LCMP is designed to communicate and act as a multi-agent CMP when a connection to the central CMP is severed, but other LCMPs can still communicate with each other. This allows context communication between devices to make informed decisions based on all available information. The lightweight systems can additionally be customised to function as individual roles within the multi-agent context system (a device that specialises in context providing may include extra storage to enable caching when disconnected).

To obtain the described functionality, the LCMA will have the following requirements:

- Interface with local context providers and consumers
- Send and receive context from a centralised CMP
- Receive instructions from a wider CMP
- Operate as an isolated CMP when disconnected from other systems
- Reconnect to centralised CMP when/if a connection is restored
- Communicate context with other deployed context systems
- Customise operation for deployment specific operation.

The design of the LCMP system is outlined in Figure 6. The communication module handles all communication of context information, either to the centralised CMP or other LCMPs in the deployment. This module also handles incoming communications of context, either for caching or for consumer control. The local context management module focuses on managing context providers connected to the LCMP, while the consumer control module focuses on managing the requests of context consumers.

Context Management Platform System Architecture - Low Level.png

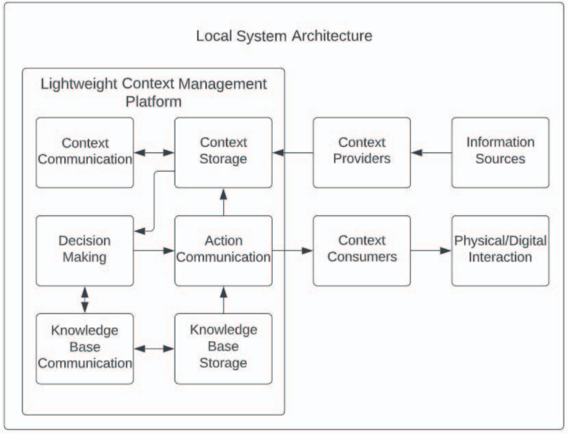

Fig. 6. Local Lightweight CMP low level design.

## V. Conclusion

While context management systems are well suited to various applications, network quality still creates problems for larger scale applications. The problems are amplified in environments where networks encounter repeated disruptions and disconnections. The operating environment faced by SAR organisations is hazardous. The development of the proposed LCMA will enable a continuous and predictable operational standard even while the connection to a centralised CMP is intermittent.

## References

[1] A. Al-alshuhai and F. Siewe, *An extension of the use case diagram to model context-aware applications*, in 2015 SAI Intelligent Systems Conference (IntelliSys), 2015, pp. 884–888.

[2] E. Badidi, *A context broker federation for qoc-driven selection of cloud-based context services*, in The 9th International Conference for Internet Technology and Secured Transactions (ICITST-2014), 2014, pp. 185–190.

[3] R. Ballamajalu, S. V. R. Anand, and M. Hegde, *Co-ioam: In-situ telemetry metadata transport for resource constrained networks within ietf standards framework*, in 2018 10th International Conference on Communication Systems & Networks (COMSNETS), 2018, pp. 573–576.

[4] Y. Banday, G. Mohammad Rather, and G. R. Begh, *Effect of atmospheric absorption on millimetre wave frequencies for 5g cellular networks*, IET Communications, 13 (2019), pp. 265–270.

[5] A. S. da Silva, P. Smith, A. Mauthe, and A. Schaeffer-Filho, *Resilience support in software-defined networking: A survey*, Computer Networks, 92 (2015), pp. 189–207.

[6] P. S. Gandodhar and S. M. Chaware, *Context aware computing systems: A survey*, in 2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2018 2nd International Conference on, 2018, pp. 605–608.

[7] L. Gong, X. Xu, Z. Du, and T. Jiang, *Research on modeling of signal field strength received by sea skimming vehicle*, in 2022 IEEE 10th International Conference on Computer Science and Network Technology (ICCSNT), 2022, pp. 149–151.

[8] J. Heer, A. Newberger, C. Beckmann, and J. I. Hong, *liquid: Context-aware distributed queries*, in UbiComp 2003: Ubiquitous Computing, A. K. Dey, A. Schmidt, and J. F. McCarthy, eds., Berlin, Heidelberg, 2003, Springer Berlin Heidelberg, pp. 140–148.

[9] G. Judd, R. Lorke, P. Boyd, V. Radenovic, and K. Chan, *Upping the iq of army's digital communications improving tactical situational awareness and command and control using semantically managed autonomous and resilient tactical networking (smartnet)*.

[10] N. Kara, M. El Barachi, A. El Bardai, and O. Alfandi, *A new business model and architecture for context-aware applications provisioning in the cloud*, in 2014 6th International Conference on New Technologies, Mobility and Security (NTMS), 2014, pp. 1–5.

[11] H. S. Khargharia, P. P. Jayaraman, A. Banerjee, A. Zaslavsky, A. Hassani, A. Abken, and A. Kumar, *Probabilistic analysis of context caching in internet of things applications*, in 2022 IEEE International Conference on Services Computing (SCC), 2022, pp. 93–103.

[12] S. Kiani, M. Riaz, Y. Zhung, S. Lee, and Y.-K. Lee, *A distributed middleware solution for context awareness in ubiquitous systems*, in 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05), 2005, pp. 451–454.

[13] J. Y. Kim and G. M. Lee, *Context awareness for smart ubiquitous networks*, in 2014 International Conference on Electronics, Information and Communications (ICEIC), 2014, pp. 1–2.

[14] W. Lee, J. Lee, and Y. Kim, *Contextual imputation with missing sequence of eeg signals using generative adversarial networks*, IEEE Access, 9 (2021), pp. 151753–151765.

[15] X. Li, M. Eckert, J.-F. Martinez, and G. Rubio, *Context aware middleware architectures: Survey and challenges*, Sensors, 15 (2015), pp. 20570–20607.

[16] P. Makris, D. N. Skoutas, and C. Skianis, *A survey on context-aware mobile and wireless networking: On networking and computing environments' integration*, IEEE Communications Surveys & Tutorials, 15 (2013), pp. 362–386.

[17] A. Nawaz, S. Ahmed, A. Ishtiaq, V. Akre, M. Taimur Ali, and S. Hameed, *Context-aware frameworks and architectures-a comprehensive survey*, in 2022 Advances in Science and Engineering Technology International Conferences (ASET), 2022, pp. 1–6.

[18] A. Neto, S. Sargento, F. C. Pinto, and E. Logota, *Context-aware session and network control in future internet*, in 2009 IEEE International Conference on Communications Workshops, 2009, pp. 1–6.

[19] B. Rauf, H. Abbas, A. M. Sheri, W. Iqbal, Y. A. Bangash, M. Daneshmand, and M. F. Amjad, *Cacs: A context-aware and anonymous communication framework for an enterprise network using sdn*, IEEE Internet of Things Journal, 9 (2022), pp. 11725–11736.

[20] E. Sarkar, E. Chielle, G. Gürsoy, O. Mazonka, M. Gerstein, and M. Maniatakos, *Fast and scalable private genotype imputation using machine learning and partially homomorphic encryption*, IEEE Access, 9 (2021), pp. 93097–93110.

[21] M. Shyama and A. S. Pillai, *Fault tolerance strategies for wireless sensor networks – a comprehensive survey*, in 2018 3rd International Conference on Inventive Computation Technologies (ICICT), 2018, pp. 707–711.

[22] S. Unni, D. Raj, K. Sasidhar, and S. Rao, *Performance measurement and analysis of long range wi-fi network for over-the-sea communication*, in 2015 13th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt), 2015, pp. 36–41.

[23] M. Vladoiu and Z. Constantinescu, *Learning with a context-aware multiagent system*, in 9th RoEduNet IEEE International Conference, 2010, pp. 368–373.

[24] M. Zafar, B. Moltchanov, and N. Baker, *Distributed context management: Architecture & commercial trials*, in 2010 Future Network & Mobile Summit, 2010, pp. 1–8.