Penerbit Akademia Baru

Journal of Advanced Research in Computing and Applications

Journal homepage: www.akademiabaru.com/arca.html
ISSN: 2462-1927

A Survivable Internet of Things Scheme

Open Access

Omar Alhazmi[1,*]

[1] Department of Computer Science, College of Computer Science and Engineering, Taibah University, Medina 30001, Saudi Arabia

ARTICLE INFO

Article history:
Received 7 April 2018
Received in revised form 2 June 2018
Accepted 10 July 2018
Available online 6 August 2018

Keywords:
Survivability, internet of things, IoT, IoE, reliability

ABSTRACT

Availability is an essential property of information systems, especially in critical infrastructure and revenue-generating systems. Availability can be compromised when an incident or an attack causes a system to shut down. Then, recovery can be applied to restore availability. One other way to achieve high availability is to start with building survivable systems. Survivable systems resist shutting down as they are built to continue operation despite being affected by incidents. Survivability depends on many factors including architecture and redundancy. Internet of Things systems are complex and often widespread, and to ensure survivability certain measures should be in place to face the challenges inherent in the nature of these systems. In this work, we preview some of the well-known IoT architectures and propose architectural schemes to ensure survivability.

1. Introduction

Ubiquitous computing emerged in the 1980s with a focus on embedding technology into the background of everyday life. An ambitious futuristic vision at the time included applications such as smart homes and smart cities. The main challenges early on were the high price of computer components and high connectivity requirements for such environments. However, both challenges no longer exist with affordable network-enabled devices and wireless network technologies. In 2010, Sundmaeker et al. defined ubiquitous computing as “the physical world that is richly and invisibly interwoven with sensors, actuators, displays, and computational elements, embedded seamlessly in the everyday objects of our lives, and connected through a continuous network” [2]. On the other hand, an extended concept of IoT is referred to as Internet of Everything (IoE) is defined by Cisco as “is the intelligent connection of people, process, data and things” [2]. The realization of IoT and IoE became feasible with the growth of the Internet and the advancement of fast, reliable wired and wireless networking. Ubiquitous computing as a paradigm is now slowly being replaced by the Internet of Things (IoT) paradigm, a term first coined by Kevin Ashton in 1999 in the context of supply

[*] *Corresponding author.*
E-mail address: Omar Alhazmi (ohhazmi@taibahu.edu.sa)

chain management [3]. Moreover, a study by Gartner suggested that the IoT Platform is an innovation trigger that will be in the mainstream within five to ten years (see Fig. **1**) [4].

For a distributed environment such as IoT, it is important to increase its resilience, robustness, and reliability. Hence, if we can at least guarantee continuous operation of most essential services, then we can consider the system as survivable. Mead defines survivability as “the capability of a system to complete its mission promptly, even if significant portions are compromised by attack or accident” [5]. Here, we propose an architecture that can guarantee a high-level of survivability with feasible resource allocation.

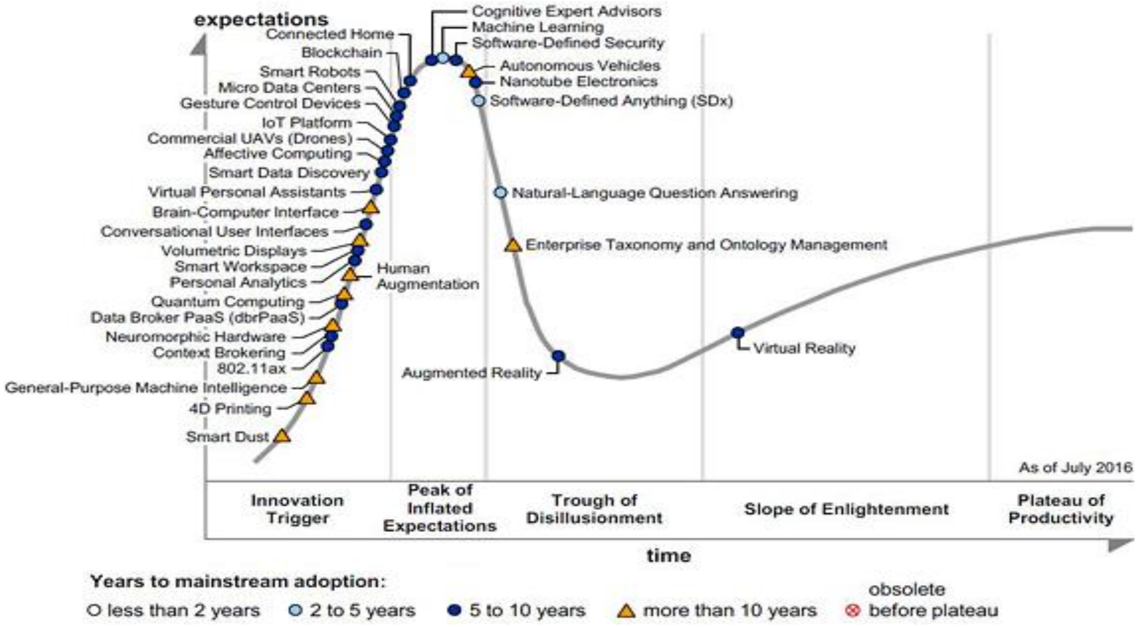

Fig. 1. Gartner Technology Mainstream Adaptation Projection [4]

2. Internet of things Architecture

IoT is still evolving with lack of standard agreed upon architecture. When researching about existing architectures, it is challenging to agree on a solution as both academia and industry offer several architectures are proposed. For example, a three-layered architecture was and a more detailed IoT architecture with five-layers presented in [6-10]. The additional layers further divide an IoT architecture to separate some of the functionality and flow of data by introducing business layer and middleware layer. Alternatively, Ning and Wang proposed an approach analogous to a human nervous system with components compared to the brain, nerve network, and sensory inputs [11], each of which will be introduced in the following sections.

2.1 Man Like Neural Network Model (MLN) Architecture

The man like neural network (MLN) model architecture (see Fig**. 2**), as proposed by Ning and Wang [11], is inspired by the layers of information processing in the human brain. The brain is the central intelligence of the system that includes the storage, processing, control, and decision-making. The second component is the spinal cord, which is analogous to the distributed network of data processing nodes of smart gateways. The third is the network of nerves, which corresponds to the networking components and sensors.

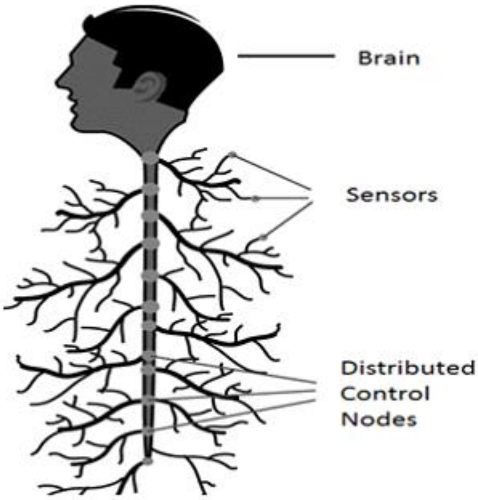

Fig. 2. The man like neural network (MLN) model architecture [11]

2.2 Three-layer Architecture

The three-layer architecture as described in [6-10] was introduced in the early stages of research in this area. It is shown in Figure 3 and consists of the perception, network, and application layers.

1. The perception layer is the physical layer, which includes devices such as sensors (e.g., for measuring environmental parameters) and actuators (to perform some action) as well as other smart devices.
2. The network layer is responsible for connecting to other smart components, network devices, and servers. It is also used for transmitting and processing sensor data.
3. The application layer provides the end user interface for the system. Through this layer, the user can benefit from the provided systems such as smart homes, smart cities, smart health, environment and weather monitoring, and other IoT applications.

The three-layer architecture is often considered an over-simplification architecture of IoT as it hides many details. Therefore, here we will primarily consider the five-layer IoT.

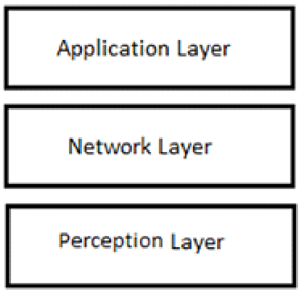

Fig. 3. Three-layer IoT Architecture

2.3 Five-layer Architecture

The five-layer architecture includes the same previous three layers along with the middleware and business layers as described in [6-10]. In order of the architecture, the layers are designed as the business, application, middleware, network, and perception layers (see Figure 4). The roles of the perception, network, and application layers are the same as with the three-layer architecture. The middleware layer is split from the network layer with the responsibility to store and analyze data, while the network layer remains responsible for transferring data from the perception to middleware layers and vice-versa. The business layer manages the entire IoT system, it is concerned with the integration between applications and business processes and enterprise systems.

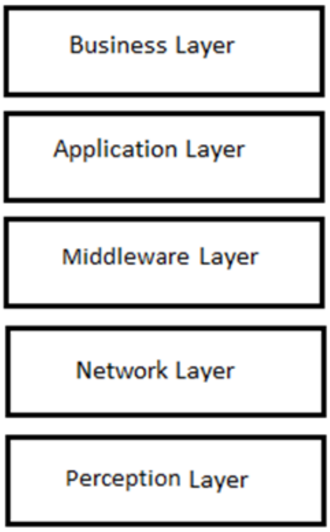

Fig. 4. Five-layer IoT Architecture

1. The business layer is concerned with integrating the applications with the enterprise system to achieve business goals, As this is less technical, disaster recovery efforts will not be addressed directly by this layer.
2. In the application layer, the user interacts with the interface to operate and monitor the entire system. The disaster recovery processing at this layer is concerned with maintaining a connection with the lower layer (middleware) and keeping the interface up and running.
3. The middleware layer is considered the core layer and can be a cloud or conventional data center. The layer hosts applications that are critical to providing services and manages the end-to-end IoT architecture. Therefore, if this layer fails, then the entire IoT system fails, as all components depend heavily on this layer handing most of the processing and data storage. The presented schemes are mainly concerned with this layer, they will be illustrated in the next section.
4. The network layer is like a conventional network. However, the traffic profile can be different due to the type and size of data being transferred. For example, some data will be transferred in a one-way direction from the sensors, readers, and RFID devices through this layer to the middleware layer. Smarter devices will feature a two-way communication through this layer. Therefore, the network topology must be considered for disaster recovery and network survivability. Moreover, it must support different protocols due to the variety of devices being used.

5. The perception layer is the most distinguishable from conventional environments consisting of embedded systems, sensors, and actuators. These are small devices with different operating systems, CPU types, and memory, so are expected to be inexpensive, single-function devices with rudimentary network connectivity, such as a temperature or pressure sensor.

3. Implenting Disaster Recovery on the Five-Layer Internet of Things Architecture

The scheme is to implement an Internet of things (IoT) environment with distributed nodes, where pairs of nodes work as a team (e.g., they host each other as a backup for disaster recovery purposes). With this scheme, each node is equipped with sufficient similar computing, storage, and communication capabilities. In each pair of nodes, the first node contains local storage while backup cloud storage serves the second node. In other words, while the second node contains cloud storage, local backup storage with disaster recovery capability also serves the first node (see Figure 5). Hence, when a disaster occurs, and some nodes become unreachable, the standby nodes take over and execute processes instead of the dead nodes.

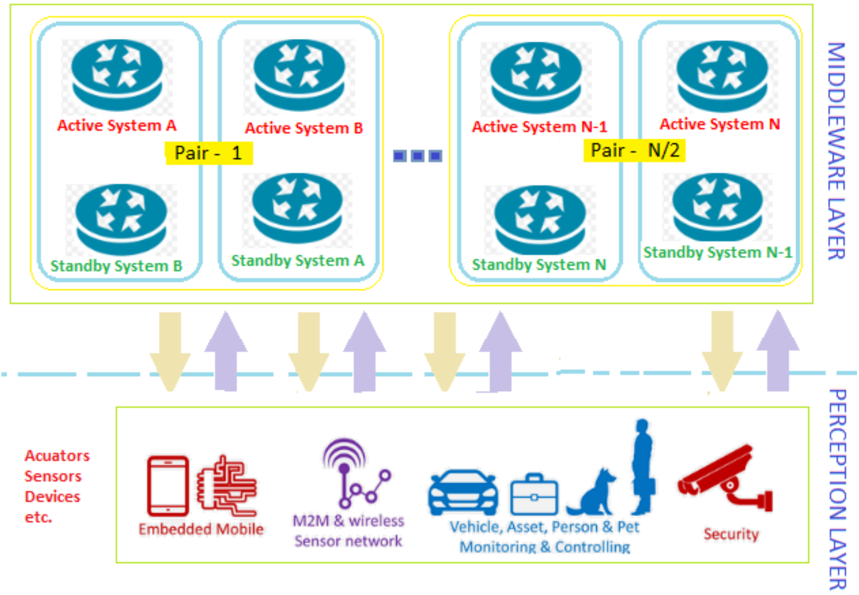

Fig. 5. IoT Environment with Middleware/Network/Perception Layers Implementation

The key advantage of this implementation is that a failure of one node will not bring the entire system down. Moreover, failure of one system will not harm the overall performance. In fact, because nodes are logically grouped in pairs, even with the failure of up to half of the system, they will still provide 100% performance in the best-case scenario. This implementation is suitable for an IoT with distributed systems where each system is close to its actuators, sensors, and devices. Each node hosts a backup of another node, and both nodes co-host each other’s backup storage.

The system can be modified with half of the nodes implemented virtually in the cloud while the other half remains as physical nodes (see Figure 6). This approach will further improve resiliency since, if the cloud fails, all virtual nodes will have a physical backup with its peer node. At the same time, failure of physical nodes will not affect the entire system as they have virtual backups in the cloud.

The system can be further modified to have all the nodes implemented virtually in the cloud (see Figure 7). This will improve resiliency, scalability, and inherit all the benefits of the pay-as-you-go model from a cloud service. Also, this scheme will benefit from the cloud provider reliability and Service Level Agreement (SLA) terms. However, the load from the communication in this environment requires larger bandwidth and costs may increase due to communication charges.

Figure 8 shows the impact on the entire system as nodes fail. As expected, when one node fails, and its backup survives, the entire system survives at 100%. Moreover, when more nodes fail, the entire system may fail in different scenarios. Figure 8 presents three possible scenarios. The best case is when all failed nodes have surviving backups. The system withstands the loss of up to 50% of its nodes with 100% functionality until a point where the trend falls sharply until all nodes fail and the system shuts down completely. The worst case is when each node fails along with its backup and the system loses 100% functionality quickly. An average case is also included in Figure 8 running between the best and worst scenarios.

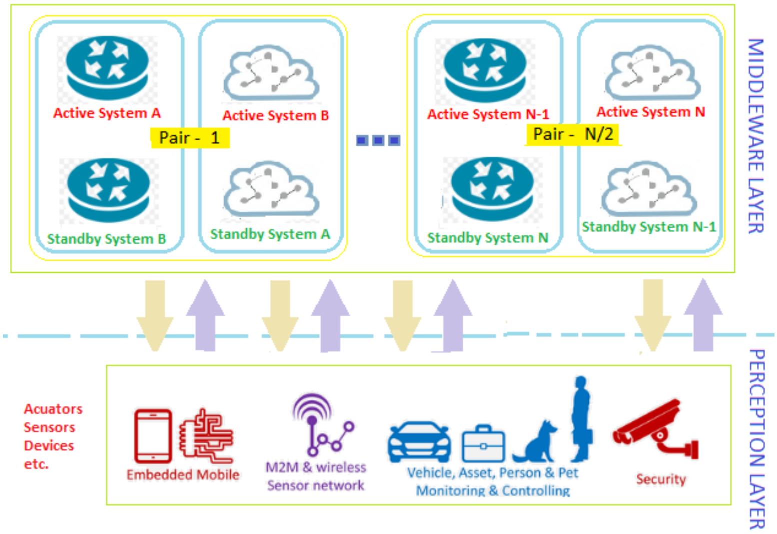

Fig. 6. IoT Environment of Middleware/Network/Perception Layers Implementation (with the cloud)

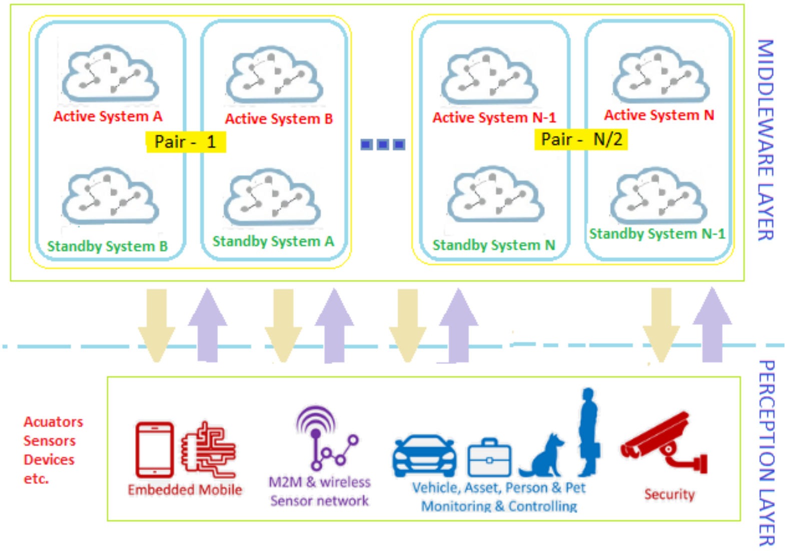

Fig. 7. IoT Environment with Middleware/Network/Perception Layers Implementation (entirely cloud-based)

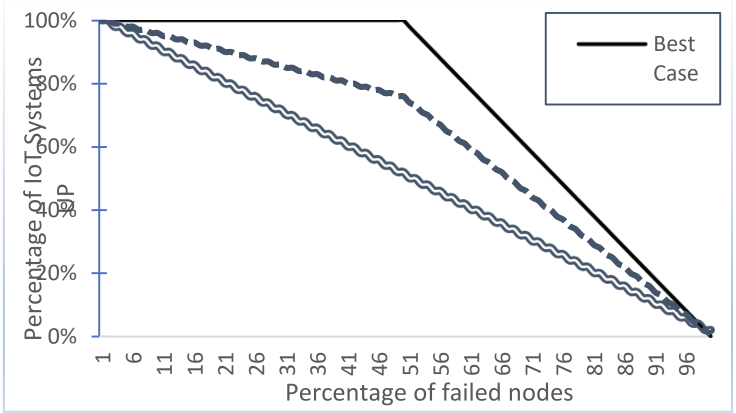

Fig. 8. Failure of nodes vs. Failure of the entire system

4. Conclusion

In this paper, we overviewed the common architectures for the Internet of Things (IoT), including the Man Like Neural Network Model (MLN) Architecture, the three-layered architecture, and the five-layered architecture. We discuss a scheme for the middleware layer in the five-layered architecture that groups the nodes (systems) into pairs. For each pair, the two nodes co-host each other's backup. We also previewed hybrid architecture featuring half-cloud, half-physical, and finally we have previewed full-cloud schemes. We demonstrated (as shown in Figure 8) that in the best-case scenario the system would have zero percent failure as nodes fail even when at the point when half the nodes fail. On the other hand, even in the worst-case scenario, the system will fail 50% slower using this scheme. Moreover, the average case will be somewhere in between. As this is a preliminary study, in future, we will compare the efficiency of the three schemes. Furthermore, other architecture can be investigated to improve survivability and to increase resilience of IoT systems.

Future work, can start to experiment with hypervisors to manage IoT devices and the disaster recovery and how this can improves RTO and RPO. Besides, hypervisors can suggest alternative architecture that is worth investigating. Also, cloud based architectures impact on security issues should be evaluated as discussed by Thiab and Shibghatullah [12].

References

[1] H. Sundmaeker, P. Guillemin, P. Friess, S. Woelfflé. "Vision and challenges for realizing the of Things," Cluster of European Research Projects on the Internet of Things - CERP IoT, 2010.

[2] Ashton, Kevin. "That 'internet of things' thing." *RFID journal*22, no. 7 (2009): 97-114.

[3] Cisco, Internet of Everything Assessment, http://ioeassessment.cisco.com/learn/ioe-faq (Accessed 18-3-2018).

[4] Gartner, Gartner's 2016 Hype Cycle for Emerging Technologies Identifies Three Key Trends That Organizations Must Track to Gain Competitive Advantage (2016) http://www.gartner.com/newsroom/id/3412017

[5] Mead, Nancy R. "Requirements Engineering for Survivable Systems." (2003).

[6] Sethi, Pallavi, and Smruti R. Sarangi. "Internet of things: architectures, protocols, and applications." *Journal of Electrical and Computer Engineering* 2017 (2017).

[7] Gubbi, Jayavardhana, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. "Internet of Things (IoT): A vision, architectural elements, and future directions." *Future generation computer systems* 29, no. 7 (2013): 1645-1660.

[8] Mashal, Ibrahim, Osama Alsaryrah, Tein-Yaw Chung, Cheng-Zen Yang, Wen-Hsing Kuo, and Dharma P. Agrawal. "Choices for interaction with things on Internet and underlying issues." *Ad Hoc Networks* 28 (2015): 68-90.

[9] Said, Omar, and Mehedi Masud. "Towards internet of things: Survey and future vision." *International Journal of Computer Networks* 5, no. 1 (2013): 1-17.

[10] Khan, Rafiullah, Sarmad Ullah Khan, Rifaqat Zaheer, and Shahid Khan. "Future internet: the internet of things architecture, possible applications and key challenges." In *Frontiers of Information Technology (FIT), 2012 10th International Conference on*, pp. 257-260. IEEE, 2012.

[11] Ning, Huansheng, and Ziou Wang. "Future internet of things architecture: like mankind neural system or social organization framework?." *IEEE Communications Letters* 15, no. 4 (2011): 461-463.

[12] A. S. Thiab, A. S. Shibghatullah , Hypervisor security issues in cloud computing: the need to mitigate the risks, *Journal of Advanced Research in Computing and Applications* 1, no. 1, (2015), 1-5.