AWS Architecture for Smart Livestock

Subjects: Agriculture, Dairy & Animal Science

Contributor: Kristina Dineva

In the ecological future of the planet, intelligent agriculture relies on CPS and IoT to free up human resources and increase production efficiency. Due to the growing number of connected IoT devices, the maximum scalability capacity, and available computing power of the existing architectural frameworks will be reached. This necessitates finding a solution that meets the continuously growing demands in smart farming. Cloud-based IoT solutions are achieving increasingly high popularity. The aim of this study was to design a scalable cloud-based architecture for a smart livestock monitoring system following Agile methodology and featuring environmental monitoring, health, growth, behaviour, reproduction, emotional state, and stress levels of animals. The AWS services used, and their specific tasks related to the proposed architecture are explained in detail. A stress test was performed to prove the data ingesting and processing capability of the proposed architecture. Experimental results proved that the proposed architecture using AWS automated scaling mechanisms and IoT devices are fully capable of processing the growing amount of data, which in turn allow for meeting the required needs of the constantly expanding number of CPS systems.

Keywords: Internet of things (IoT) ; Amazon Web Services (AWS) ; Smart Livestock farm

1. Introduction

There is a steady trend in increasing the number of animals on a farm. This growth in number varies around the world depending on the agriculture type and social structure. In Australia, there are on average 279 cows per farm, while in New Zealand, they average 440 [1][2]. Farms in Europe that keep more than 1000 animals such as farms in Estonia [3] are also becoming more common. An analogous situation is observed in the United States, wherein Michigan, Ohio, Indiana, and other traditional dairy states in the East and Midwest are built farms with upwards of 1000–5000 animals [4].

It has been found that the increase in herd size is driven by economies of scale—the cost of production per unit decreases with increasing herd size [5]. As pointed out in [5], the association between herd size and health and welfare is complex and affected by many factors. Providing farmers with access to rich data sources can aid in improving animal health and welfare. However, to achieve these goals, farmers must adopt modern technologies with an increased level of integration of automation processes and software management.

The development of technologies forces the transition to Agriculture 5.0 [6], where through automation and the introduction of recent technological solutions, traditional farm practices are modified and improved. As there is a constant need for development and due to the increasing trend in the number of animals in a herd/farm, software architectures need to be easily scaled to be able to receive a lot of data simultaneously from numerous Internet of things (IoT) devices. On the other hand, the current trend of increasing the urban population at the expense of people in rural areas raises the problem of reducing human participation in animal husbandry and the inclusion of more technological solutions to ensure humane breeding and careful monitoring of animal welfare. The advanced technological solutions led to the creation of smart farm systems using the communication capabilities of distributed and interconnected computing devices. Systems, where physical objects are represented in the digital world and integrated with computation, storage, communication capabilities, and are connected to each other in a network, can be defined as cyber–physical systems (CPS) [7]. Internet of Things technologies serves to integrate cyber–physical systems and to be an interface between CPS and their users. IoT is based on the idea of establishing a permanent connection between the physical and digital worlds [8]. According to ISO/IEC JTC1 2015, both terms, IoT and CPS, are interchangeable [9]. Both CPS and the Internet of Things are used to free up human resources, increase modern production efficiency, and be of significant help to improve product quality in smart agriculture [10]. It uses CPS and IoT technologies to increase productivity in smart farms through modern means in a continuously sustainable way to achieve the best in terms of quality, quantity, and financial return as well as to ensure that the process is environmentally friendly [11].

Different approaches for the design of software architectures in the field of livestock have been proposed. In [6], the usage of remote sensing (RS) technology in agriculture is discussed. It was pointed out that “the RS technology generates huge sets of data that necessitate the incorporation of artificial intelligence (AI) and big data to extract useful products, thereby augmenting the adeptness and efficiency of agriculture to ensure its sustainability" [12]. The reference architecture proposed in [13] has several layers and provides a model for farm management information systems. However, "the complete versions of the feature diagrams as well the detailed implemented architecture designs have not been shown" [13]. The study [14] discusses Global Sensor Network (GSN) as a foundation for the management of the streaming sensor data. In the proposed architecture, the signals received from the IoT sensors are collected by a gateway located on the farm and sent to smart farm servers through a high-speed broadband network.

It is expected that IoT technology can make a breakthrough in livestock management by connecting the biological information of livestock and environmental information obtained by IoT sensors to farmers who are in a remote location on their farms via the cloud [15][16]. Attention was paid to the design of systems with analytical intelligence capability and data to be present on the premises. Systems for monitoring animals are available to farmers. Some of the approaches that are used in the design process are "domain-driven" [13]. The idea behind these was to make software development easier by providing a model for building flexible and reusable applications [17]. However, building the required infrastructure for a complex multi-layer architecture is very time consuming and, in many cases, is considered as an anti-pattern [18] and it is wise to just use another architectural approach.

An additional concept is the shifting from computing with centralised servers to distributed ones. One of these distributed technical infrastructures is presented by microservice architecture (MSA). The developed microservice architecture of a distributed IoT system discussed in detail in [19] consists of a group of microservices that communicate with each other synchronously or asynchronously.

Almost no current livestock systems extensively use the monitoring and security updates of remote IoT devices, which is a vital part of the system. Data security and sovereignty is essential nowadays as devices are exposed to different threats [20]. These aspects are deeply covered and explained in [21]. However, there is still a major problem to be addressed, providing centralised IoT device management where all remote location IoT devices can be configured remotely, and security patches applied as soon as the need arises.

Due to the growing number of connected IoT devices, the multi-layer and MSA architecture's scalability capacities and the available computing power limits are about to be reached. As many IoT devices are intended for use in the smart livestock system and will generate large volumes of heterogeneous data with high variety and roughness, this imposes the need to find a solution that meets the constantly growing requirements. There is a need to use cloud computing to ensure a reliable and secure infrastructure that supports automatic scaling of resources according to the system needs and centralised IoT device management. Cloud-based Internet of Things (CB-IoT) [22] is becoming an increasingly popular and desirable solution.

Recently, an interesting project was developed using CB-IoT by Amazon Web Services (AWS) for cattle management [23]. Within this project, the Australian team proposed a "Ceres Tag Management System" that houses the data and metadata on each ear cattle tag. The ear tag connects via satellite for data transmission, enabling the traceability of that tag throughout each animal's life cycle. The proposed architecture focuses on data collection and transmission. The analytical processing of these data is outsourced. However, this project demonstrates that the use of cloud computing in this monitoring system is essential for achieving high speed, accuracy, and security in the data processing.

2. Current Insight on IoT Architecture Based on AWS for Smart Livestock

2.1. Used Services in AWS Cloud

The proposed architecture designed for smart livestock farming is shown in **Figure 2**. It was built with AWS serverless services and contains several groups of services such as compute, storage, databases, analytics, network, mobile, management, developer, IoT, security, enterprise application, and others. The necessary services for creating a scalable and robust livestock monitoring system were grouped in frames. These were: AWS IoT Core, Lambdas, Data Recognition, Streaming Data, Data Lake, Data Stores, Data Processing, Machine Learning, Notifications, Analytics, Logging, and User Identities. The frames were developed to satisfy the required functionalities described in the Materials and Methods section as Stage 2. The designed architecture complies with all pillars of the well-designed architecture described in the Materials and Methods section on Stage 3.

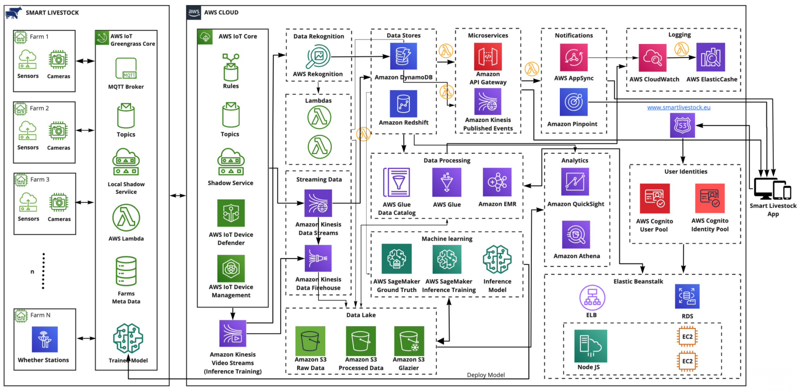

Figure 2. Architecture design for livestock smart farming. (https://www.smartlivestock.eu/, accessed on 20 May 2021).

2.2. AWS IoT Core Frame

The AWS IoT core consists of five services that maintain the needs of all IoT devices, connect to AWS cloud, manage devices, update over-the-air (OTA) [24], and secure the IoT devices. It uses the TLS communication protocol to encrypt all communication [25]. Services in this framework are rules, topics, shadow service, AWS IoT device defender, and AWS IoT device management.

The Rules enable the IoT devices developed for smart livestock to interact with AWS services.

Some of the rules used in the system are:

– Filtering data coming from IoT devices;

– Separation and recording of data according to their type in various kinds of databases;

– Sending notifications to users in certain circumstances (for example, occurrence of abnormal events in the monitoring process);

– Real-time processing of messages coming from a large number of IoT devices from different locations;

– Setting alarms to notify the user when reaching predefined limits of certain parameters (for example, reaching a critical battery level of IoT devices);

– Send the data from a Message Queuing Telemetry Transport (MQTT) message to a machine learning frame to make predictions based on the ML model;

– Send message data to an AWS IoT analytics channel; and

– Send data to a web application.

AWS core default implementation is based on the MQTT protocol, which is used by AWS core to interact with devices. It decouples the producer and consumer by letting clients publish and having the broker decide where to route and copy messages. Rules are analysed and actions are performed based on the MQTT topic stream. The Topics identify AWS IoT messages. A message broker is used to apply topic names and topic filters to route messages send using MQTT and Hypertext Transfer Protocol (HTTP) to the Hypertext Transfer Protocol Secure (HTTPS) message URL. First, it is important to create a hierarchy of topic names that identify the relevant IoT system. Topic names and topic filters are UTF-8 encoded strings. A topic name could refer to a sensor in developed IoT devices (for example, sensor/temperature/caw1/farm1). After forming hierarchical identifying topic names, topic filters are created, which filter the messages by topic names and send them to the services subscribed to them.

As many IoT deployments consist of hundreds or thousands to millions of devices, it is essential to indicate (shadow) IoT devices. AWS shadow service creates a shadow identification of each IoT device used in the smart livestock frame. In this way, each of the used devices can be accessed and managed specifically by different services. These changes are made via the MQTT protocol or via HTTP using the device shadow REST API [26] (p. 525).

Due to the specifics of the used IoT devices in this application, they have limited computation, memory, and storage capabilities. This restricts opportunities for security. Therefore, additional measures are required to ensure the security of the devices. AWS IoT device defender [27] (p. 31) was chosen in this application. Registering the IoT livestock device with an AWS IoT core provides the digital identity of the IoT livestock device in the cloud. The IoT livestock devices use the certificates to identify themselves to the AWS IoT core. These certificates are generated by the AWS IoT core. From this moment, the AWS IoT device is responsible for maintaining and implementing IoT configurations such as device identity, device authentication, device authorisation, and device data encryption. This is an out-of-the-box solution as this service provides the necessary protection for all devices by constantly checking for deviations of IoT configurations and the occurrence of abnormal deviations from the expected behaviour of the device. In the case of abnormal behaviour detection, the user and CloudWatch service are immediately informed.

AWS IoT device management is a service that remotely manages IoT devices (individually or in groups) when a signal is detected for abnormal behaviour in the devices. This service eliminates problems by managing and updating the software and firmware.

2.3. Lambda Frame

A Lambda function is a stateless piece of code, with input and output that can be triggered from a wide array of sources internal and external to AWS. It can be used to automatically scale an application up and down without making capacity planning decisions. Unlike an EC2 instance [28], a Lambda has one dedicated purpose and intentionally only runs for up to a few minutes. Lambda functions scale instantly to hundreds of instances, with almost no platform maintenance.

In this work, Lambda functions in the livestock monitoring system are responsible for:

– Converting each frame to jpeg format if it is different;

– Storing raw frames in Amazon S3 by generating a unique key for each frame, which contains specific data about the place and time of the shooting: frames/farm_1/year/month/day/hour/min/sec/frame_number.jpeg;

– Transmitting frames in jpeg format to Amazon "Rekognition" frame;

– For each frame, Lambda functions save to Amazon DynamoDB specific metadata such as a unique ID, S3 bucket and key, where the frame is saved, the approximate recording time and more.

2.4. "Data Rekognition" Frame

"Data Rekognition" framework deals with data coming from the cameras in the smart livestock IoT system. Each recorded video is divided into separate frames so that they can be processed and analysed in near real-time.

AWS Rekognition Service analyses objects, people, animals, scenes, and activities contained in the collected videos. Through this service, the application performs counting and identification of the presence or absence of specific animals, people, or objects around the farm in real-time. Personal protective equipment (PPE) detection is also performed. The results of the performed analyses are sent for storage in NoSQL database service DynamoDB, from wherein the presence of certain abnormal phenomena (identification of a missing animal, abnormal behaviour of an animal, or the presence of a stranger on the farm) are sent to Amazon Pinpoint via Kinesis Published Events.

AWS Rekognition is configured to detect and recognize faces in streaming video and to detect and count livestock. It uses AWS Kinesis Video Streams to receive and process a video stream, which is H.264 encoded. Used events are face detection and face comparison, and livestock label detection.

AWS Rekognition is used currently for several cases, and different performances have been observed. It has high accuracy when identifying persons, counting animals entering, or leaving an area one by one. However, the accuracy drops significantly for animal detection, especially in the part where the animal head position (high/low) is of interest. The results are also influenced by the livestock density occupying the same areas.

2.5. Notification Frame

Amazon Pinpoint [27] (p. 49), as a communication service, connects with users through various channels such as email, SMS, push, or voice. This service is used to personalise messages with the right content. It sends push notifications to the smart livestock application after pre-provided data that authorises PinPoint to send messages. The credentials that are provided depend on the operating system:

– For iOS apps, an SSL certificate is provided. The certificate authorises the PinPoint service for sending messages to the smart livestock apps.

– For Android apps, a web API key is provided. These credentials authorise the PinPoint service for sending messages to the smart livestock apps.

AWS AppSync takes care of managing and updating real-time data between web and mobile app and cloud. Additionally, it allows apps to interact with data on mobile devices when it is offline.

References

1. Dairy Australia, Report: In Focus 2020, The Australian Dairy Industry. 2020, pp. 6–7. Available online: https://www.dairyaustralia.com.au/industry-statistics/industry-reports/australian-dairy-industry-in-focus (accessed on 20 May 2021).
2. Livestock Improvement Corporation Limited & DairyNZ, 2019-20 New Zealand Dairy Statistics. 2020, pp. 5–8. Available online: https://www.lic.co.nz/about/dairy-statistics/ (accessed on 19 May 2021).
3. DG Health and Food Safety, Overview Report: Welfare of Cattle on Dairy Farms. 2017, p. 18. Available online: https://op.europa.eu/en/publication-detail/-/publication/8950fa88-d651-11e7-a506-01aa75ed71a1 (accessed on 20 May 2021).
4. MacDonald, M.J.; O'Donoghue, J.E.; McBride, D.W.; Nehring, F.R.; Sandretto, L.C.; Mosheim, R. Profits, Costs, and the Changing Structure of Dairy Farming, Economic Research Report; United States Department of Agriculture: North Bend, WA, USA, 2007; pp. 3–5.
5. Barkema, H.; Von Keyserlingk, M.; Kastelic, J.; Lam, T.; Luby, C.; Roy, J.; LeBlanc, S.; Keefe, G.; Keltonll, D. Invited review: Changes in the dairy industry affecting dairy cattle health and welfare. J. Dairy Sci. 2015, 98, 7426–7445.
6. European Commission, Industry 5.0. Available online: https://ec.europa.eu/info/research-and-innovation/research-area/industrial-research-and-innovation/industry-50_en (accessed on 25 January 2021).
7. Tsochev, G.; Yoshinov, R. Research on Cyber-Physical Systems Security. Probl. Eng. Cybern. Robot. 2021, 75, 3–4.
8. Stoyanov, S.; Orozova, D.; Popchev, I. Internet of Things water monitoring for a smart seaside city. In Proceedings of the 20th International Symposium on Electrical Apparatus and Technologies (SIELA), Burgas, Bulgaria, 3–6 June 2018; pp. 1–3.
9. ISO/IEC JTC 1, Internet of Things (IoT). Available online: https://www.iso.org/files/live/sites/isoorg/files/developing_standards/docs/en/internet_of_things_report-jtc1.pdf (accessed on 12 February 2021).
10. Sowmya, J.; Shetty, C. IoT and Data Analytics Solution for Smart Agriculture: The Rise Fog Computing in the Digital Era; IGI Global: Pennsylvania, PA, USA, 2019; Volume 9.
11. Haris, I.; Fasching, A.; Punzenberger, L.; Grosu, R. CPS/IOT Ecosystem: Indoor Vertical Farming System. In Proceedings of the IEEE 23rd International Symposium on Consumer Technologies (ISCT), Ancona, Italy, 19–21 June 2019; pp. 47–52.
12. Martos, V.; Ahmad, A.; Cartujo, P.; Ordoñez, J. Ensuring Agricultural Sustainability through Remote Sensing in the Era of Agriculture 5.0. Appl. Sci. 2021, 11, 5911.
13. Köksal, Ö.; Tekinerdogan, B. Architecture design approach for IoT-based farm management information systems. Precision Agric. 2019, 20, 926–958.
14. Gaire, R.; Lefort, L.; Compton, M.; Falzon, G.; Lamb, D.; Taylor, K. Demonstration: Semantic Web Enabled Smart Farm with GSN. In Proceedings of the International Semantic Web Conference (Posters & Demos), Sydney, Australia, 21–25 October 2013.
15. Akhigbe, B.I.; Munir, K.; Akinade, O.; Akanbi, L.; Oyedele, L.O. IoT Technologies for Livestock Management: A Review of Present Status, Opportunities, and Future Trends. Big Data Cogn. Comput. 2021, 5, 10.
16. Goddard, M.E. Uses of genomics in livestock agriculture. In Animal Production Science; CSIRO Publishing: Clayton, Australia, 2012; pp. 73–77.
17. Kachurka, V. Design patterns in N-tier architecture. In Proceedings of the XV International PhD Workshop OWD 2013, Wisła, Poland, 19–22 October 2013.

18. Simmons, D. Entity Framework-Anti-patterns to avoid in N-tier architecture. MSDN Magazine. June 2009, Volume 24. Number 06. Available online: https://docs.microsoft.com/en-us/archive/msdn-magazine/2009/june/anti-patterns-to-avoid-in-entity-framework-n-tier-applications (accessed on 10 May 2021).
19. Dineva, K.; Atanasova, T. Architectural ML Framework for IoT Services Delivery Based on Microservices. In Distributed Computer and Communication Networks; Lecture Notes in Computer, Science; Vishnevskiy, V.M., Samouylov, K.E., Kozyrev, D.V., Eds.; Springer: Cham, Switzerland, 2020; Volume 12563.
20. Thangarasu, R.; Anandamurugan, S. Challenges and Applications of Wireless Sensor Networks in Smart Farming—A Survey; Springer: Berlin, Germany, 2019; pp. 353–361.
21. Dineva, K.; Atanasova, T. Security in IoT Systems. In Proceedings of the XIX International Multidisciplinary Scientific GeoConference SGEM, Vienna, Austria, 9–12 December 2019; Volume 19, pp. 576–577.
22. da Silva, A.F.; Ohta, R.L.; dos Santos, M.N.; Binotto, A.P.D. A Cloud-based Architecture for the Internet of Things targeting Industrial Devices Remote Monitoring and Control. IFAC-PapersOnLine 2016, 49, 108–113.
23. The Satellite Ear Tag that is Changing Cattle Management. Available online: https://aws.amazon.com/blogs/architecture/the-satellite-ear-tag-that-is-changing-cattle-management/ (accessed on 20 May 2021).
24. Villegas, M.; Orellana, C.; Astudillo, H. A study of over-the-air (OTA) update systems for CPS and IoT operating systems. In Proceedings of the 13th European Conference on Software Architecture, Paris, France, 9–13 September 2019; pp. 269–272.
25. Khandkar, S.V.; Hanawal, K.M. Masking Host Identity on Internet: Encrypted TLS/SSL Handshake, Cryptography and Security; Cornell University: Ithaca, NY, USA, 2021.
26. AWS Official Documentation, AWS IoT Core, Developer Guide. January 2018. Available online: https://docs.aws.amazon.com/iot/latest/developerguide/iot-dg.pdf#iot-device-shadows (accessed on 10 January 2021).
27. AWS Official Documentation, Overview of Amazon Web Services, Whitepaper. April 2021. Available online: https://d1.awsstatic.com/whitepapers/aws-overview.pdf (accessed on 2 May 2021).
28. Kokkinos, P.; Varvarigou, T.A.; Kretsis, A.; Soumplis, P.; Varvarigos, E.A. SuMo: Analysis and Optimization of Amazon EC2 Instances. J. Grid Comput. 2015, 13, 255–274.