OVERVIEW OF FOG COMPUTING PLATFORMS

Aygul Tumaeva

assistant teacher of department of “Digital technologies and economics”

Nukus branch of Tashkent university of information technologies named after Muhammad al-Khwarizmi, Uzbekistan

Miyasar Uzakova

3rd year student

Nukus branch of Tashkent university of information technologies named after Muhammad al-Khwarizmi, Uzbekistan

Ernazar Uzakbaev

3rd year student

Nukus branch of Tashkent university of information technologies named after Muhammad al-Khwarizmi, Uzbekistan

Abdulaziz Daryabaev

2rd year student

Nukus branch of Tashkent university of information technologies named after Muhammad al-Khwarizmi, Uzbekistan

In reviewing existing platforms for deploying fog computing, commercial platforms as well as open source platforms were considered. The complexity of the analysis of commercial platforms lies in the lack of information about their architecture and applied technical solutions, which constitute a trade secret. However, an analysis of commercial solutions showed that among commercial fog platforms, there are both platforms with full support for fog computing (computing, analytics and organization of the transport layer of the fog network), and platforms that provide only the transport layer of the fog network and do not provide management of computing nodes and actually foggy calculations on them. Platforms that provide only the transport layer of fog computing will not be considered in this paper.

The following key characteristics of private and public commercial fog platforms can be distinguished (see Tables 1–2) [1-2].

1. Supported hardware platforms – the platform can work with any device that supports virtualization or containerization, or only with a limited list of devices – through drivers or proprietary devices. Smartiply Fog,ThingWorx and Cisco IOx only work with their own hardware.

2. The underlying development technology – on the basis of which executable environment applications are created and launched.

3. Openness of communication protocols and SDK – are there any restrictions on the applications that can be used in the fog: is it necessary to port applications, or in principle only applications written using special supplied SDKs can be executed, as, for example, in the case of ThingWorx, whose fog applications must be written using the proprietary SDK to run in fog.

4. Deployment technology – which of the fog node deployment technologies is used, if known.

5. Integration options – whether there is the possibility of integration with other platforms, such as enterprise solutions or public clouds.

6. Connecting external data sources – the ability of the platform to connect to third-party databases and data storages physically located outside the central cloud for storing and processing data.

7. Availability of additional services (Machine Learning, Analytics, etc.) – the ability to connect and use additionally supplied services that provide additional functionality for analyzing and working with data in the fog.

8. Edge support – the ability to connect and use edge devices and edge computing, and further collect and process information from them.

PRIVATE FOG PLATFORMS

Private fog platforms provide the creation of private fog solutions based on computing infrastructure deployed directly on the customer’s resources.

Cisco Platform IOx was introduced by Cisco in 2014 [3-4] as a development of network infrastructure in connection with the expected growth of IoT. The main emphasis in the platform is on reducing labor costs for porting applications to fog nodes, which is achieved through the use of containerization technologies and basing your own OS based on a Linux system.

Cisco IOx is an application environment that combines Cisco IOS (a mini operating system installed on all Cisco hardware) and Linux. Applications are developed using open source Linux utilities. A single protocol for the interaction of fog applications is used throughout the network, organized using Cisco technologies IoT. Fog applications that can run on the IOx infrastructure are supplied by both Cisco and the company's partners. Applications can be developed in a variety of general-purpose programming languages [5].

Docker is used to develop and deploy applications. A variety of application types are supported, including Docker containers and virtual machines (if the network hardware has the capability). It is also possible to use the native IOx runtime to write applications in high-level programming languages (such as Python).

Nebbiolo platform Technologies is aimed at the corporate industrial market, which supports the concept of Industry 4.0 [6-8]. Nebbiolo Company Technologies works closely with Toshiba Digital Solutions [9-11] in the supply of ready-made computing solutions for the industrial and IoT sectors.

The platform consists of the fogNode hardware, the fogOS software stack, and the fogSM system manager, deployed in the cloud or on-premises. Fog System Manager (fogSM) provides a cloud-based, centralized management platform that allows devices to be deployed and configured at the edge [12].

A key feature of the platform is fogOS, a software stack that provides fog-like connectivity, data management, and application deployment. Based on a hypervisor, fogOS provides a set of features in a virtualized form. Supports a wide range of device connection standards and allows applications to be hosted and controlled in real time [13-15].

The ClearBlade platform is a technology stack that enables the rapid development and deployment of enterprise IoT solutions from edge devices to cloud services. It includes software components installed on the entire stack of IoT devices, and also provides the ability to connect third-party systems through the provided API for integration with devices, internal business applications and cloud services [16]. The ClearBlade platform provides a centralized management console for IoT applications that can be deployed both locally and in the cloud. Platform control functions are delegated to edge nodes (either on the end devices themselves or gateways to them) using the ClearBlade fog and edge computing system edge.

Table 1. Overview of private fog platforms

Characteristic	**Clearblade**	**SmartiplyFog**	**LoopEdge**	**ThingWorx**	**Nebbiolo Technologies**	**Cisco IOx**
Supported hardware platforms	Universally	Own equipment	Universally	Own equipment	Universally	Own equipment
Basic development technology	JavaScript	There is no data	Universally (docker)	Java VM	Universally (docker)	docker, Linux ,IOx
Open communication protocols and SDK	+	+	+	-	+	+
Deployment Technology	Linux KVM	There is no data	Docker	There is no data	Docker	linux KVM
Integration options	Oracle, SAP, Microsoft, sales force	-	-	Microsoft Azure IoT Hub	-	Microsoft Azure IoT Hub
Connecting external data sources	+	-	+	+	+	+
Availability of additional services	There is no data	+	-	+	+	+

Edge Support	+	+	+	+	+	+

The platform supports serverless computing is an approach to developing JavaScript -based services that can be configured to implement machine learning and data analysis methods. The platform provides mechanisms for exporting data and analytics collected by the system to widely used business systems, applications and databases through integration with enterprise platform solutions from Oracle, SAP, Microsoft and Salesforce.ClearBlade also provides proprietary dashboards, business applications, and database management systems for end-to-end surveillance and management of the IoT ecosystem [17-18].

ClearBlade uses an OAuth model for access control, where each user and device receives a token that must be authorized to gain access to a system or host. Data is encrypted both on the devices themselves and over the network. Transmitted data is encrypted using OpenSSL libraries with TLS encryption.

Smartiply Platform Fog is a foggy computing platform that focuses on optimizing resources and keeping the devices running on the system even when not connected to the cloud. For online environments, the platform provides higher reliability due to the optimization of resources and calculations, which are performed on the basis of proprietary hardware. The platform provides interaction between devices on a point-to-point basis. Thus, the system of nodes can continue autonomous work of receiving, analyzing and storing data, until the connection with the external network is restored [19-20].

LoopEdge Platform by Litmus Automation allows you to connect various devices into a single system, collect and analyze data from them. Also Litmus Automation provides a standalone Loop platform to manage the lifecycle of any IoT device and export real-time data to internal analytics and business applications.

The developers of the platform emphasize that it is able to work with almost any device, both industrial and affordable to the consumer. For example, the platform supports connecting devices based on Arduino and Raspberry Pi. Even if some device is not supported, it is quite easy to connect it to the platform, due to the executable packages installed on the device itself, which can be extended and created from scratch for a specific device [21]. All instructions are in the public domain.

This platform has become widespread among well-known engineering concerns: Nissan, Renault, Mitsubishi Corporation techno.

The PTC ThingWorx platform is an IoT platform that offers connectivity to a fairly large number of devices (drivers for 150 types of devices are available) [22]. However, due to the fact that devices are connected through a driver that requires installation before the device can be used in the fog, this platform is not universal and has limitations on the devices used.

Applications for the platform need to be written from the supplied SDKs used. Further data analysis and business process management also goes through the provided tools of the platform itself. To accomplish these tasks, the platform has an extensive section for developers with instructions and tutorials, as well as help from specialists from the company itself to install, configure and extend the platform. Also "out of the box" it is possible to connect to Microsoft Azure IoT hub.

Public fog platforms

Public fog platforms today are solutions of major players in the cloud computing market, focused on solving data processing problems from IoT systems, tied to the capabilities of the corresponding cloud platform. The key characteristics of the considered public fog platforms are given in Table. 3.

Azure platform IoT provides a platform for fog and edge computing based on the technology stack from Microsoft. Azure platform IoT consists of several large subsystems such as IoT Central, IoT Edge, which in turn base their work on Microsoft cloud technology Azure. Connecting devices from Microsoft partners is possible without the use of drivers or code using IoT technology Plug and play. This approach is possible for devices running any OS, including Linux, Android, Azure SphereOS, Windows IoT, RTOS and others [23-24].

Create, install, and manage fog apps through the Azure portal IoT hub. IoT Hub is a cloud-hosted managed service that acts as a central message handler for bi-directional communication between an IoT application and the devices it manages. IoT The Hub supports both device-to-cloud and cloud-to-device data transfer. IoT The Hub supports several messaging patterns such as device-to-cloud telemetry, device-to-device file uploads, and challenge-response technology to control devices from the cloud.

To deploy computing closer to the devices themselves or on the devices themselves, use Azure IoT Edge, which allows you to deploy applications with your own business logic or ready-made applications already in the catalog on end devices using containerization technology.

Amazon AWS IoT Platform Greengrass allows you to extend the capabilities of AWS (Amazon web Services) to edge devices, allowing them to work with data locally while using the cloud to manage, analyze and securely store data. AWS IoT Greengrass allows connected devices to run AWS Lambda functions, run Docker containers, make predictions based on machine learning models, sync device data, and communicate securely with other devices even when they're not connected to the Internet.

AWS IoT Greengrass allows you to create IoT solutions that connect different types of devices to the cloud and to each other. AWS IoT Greengrass Core can be used on Linux devices (including Ubuntu and Raspbian distributions) that support Arm or x86 architectures. AWS IoT Service Greengrass Core provides local execution of AWS Lambda code, message passing, data management, and security. Devices with AWS IoT Greengrass Cores act as service portals and can interact with other devices running FreeRTOS(Real-time operating system for microcontrollers) or have the AWS IoT Device SDK installed. The size of such devices can vary greatly, from small microcontroller-based devices to large household appliances. When a device with AWS IoT Greengrass Core loses connection to cloud, devices in AWS IoT group Greengrass can continue to communicate with each other over the local network.

The Google, Yandex and Mail.ru platforms provide their own cloud and fog solutions for collecting, storing, processing, analyzing and visualizing data. The collected data from devices is integrated into a public cloud system for deeper processing and analysis (including machine learning and artificial intelligence) due to the high computing power of the cloud. These platforms support many protocols for connecting and interacting through the provided API [25]. There are a large number of ready-to-use services available for installation in the directory of the platform itself, which can be connected to your own fog solution, combining with each other.

Table 2. Overview of public fog platforms

Characteristic	AWS Greengrass	Azure IoT	Google	Yandex	Mail.ru
Supported hardware platforms	Universally	Universally	Universally	Universally	Universally
Basic development technology	Universally (docker)	Universally (docker)	Universally	Universally	Universally
Open communication protocols and SDK	+	+	+	+	+
Deployment technology	Docker	Docker	Docker	Docker	Docker
Possibility of integration	Amazon elastic Compute 2	Azure, via API	Services Google and partners, via API	Universal via API	Universal via API
Connecting external data sources	-	-	+	+	-
Availability of additional services (Machine learning, Analytics, etc.)	+	+	+	+	+
Edge support	+	+	+	+	+

Open source fog platforms

In the course of the analysis of existing solutions, we conducted an overview of existing open source fog platforms. Unlike commercial solutions, for open source platforms provides complete descriptions of architectures, requirements for computing resources, as well as technologies used, both at the hardware and software levels (see Table 4).

Table 3. An overview of open source fog platforms

	Target	**Implementation**
FogFrame 2.0	Check Concept Model	-
FogFlow	Easier and more flexible service orchestration	+
FogBus	Overcome heterogeneity at the OS and P2P communication level of various fog nodes	-

FogFrame2.0 Framework - Open source fog frameworkcode [26], aimed at deployment on single-board computers (Raspberry Pi).It was introduced to solve the following tasks:

– define and implement the functions of the fog infrastructure, i.e. creation and maintenance of a foggy landscape (many foggy nodes and devices) and application management;

– implement heuristic algorithms for placing services in the fog, namely the primary fit algorithm and the genetic algorithm;

– introduce mechanisms for adapting to dynamic changes in fog and for recovering from overloads and failures.

To evaluate the behavior of FogFrame, various models of application request arrivals (constant, pyramid, and random traversals) were applied, as well as the processes of computing service placement were examined. The goal of the study was ultimately to observe the behavior of the platform and its response to failures. The platform responds dynamically to events at runtime, i.e. when new devices appear or go offline, when devices experience failures or overloads, the necessary relocations of nodes are performed.

The FogFlow platform is an open source fog platform. The main goal of the developers of this platform was to provide a simple and flexible way to develop, deploy and orchestrate fog services. The uniqueness of their approach lies in:

– context-sensitive orchestration of services, while other services simply operate on events (event) or groups of events (topic) occurring in the network;

– FogFlow services and applications are designed from the perspective of all cloud nodes and edge nodes, not from the perspective of each individual edge node.

FogFlow proposes a data flow programming model enhanced with declarative hints based on the widely used NGSI standard, which provides service developers with two benefits:

1) fast and easy development of fog computing applications, as such hints hide a lot of settings and deployment complexities from service developers;

2) a high degree of openness and interoperability for information exchange and integration of data sources.

FogFlow is one of the components of a large open IWARE infrastructure that provides the development and implementation of various smart solutions. This framework is one of the modern cloud frameworks along with Amazon web services. To implement and use FogFlow, a wide library of ready-made solutions from the developer community and detailed implementation instructions are available.

FogBus platform (supported by Melbourn clouds Lab) integrates various hardware tools through software components that provide structured interaction and platform-independent application execution [27]. FogBus uses blockchain to ensure data integrity when transferring sensitive data. The platform-independent architecture of application execution and interaction between nodes allows you to overcome heterogeneity in the integrated environment.

FogBus supports the implementation of various resource management and scheduling policies for executing IoT applications written using parallel programming models such as SPMD (single program, multiple data).

Sleep data analysis application prototype is used to evaluate the performance of the FogBus platform. Apnea. This example illustrates how a (healthcare) application built using the SPMD model can be implemented using various FogBus settings to process IoT data in an integrated computing environment.

This framework makes it easy to deploy IoT applications, monitor and manage resources. FogBus system services are developed in cross-platform programming languages

(PHP and Java) and are used with Extensible Application Layer Protocol (HTTP), which helps FogBus overcome heterogeneity in the OS and P2P communication layer of various fog nodes. In addition, the FogBus platform functions as a Platform as a Service (PaaS) model for the Fog integrated environment. Cloud, which not only helps application developers create various types of IoT applications, but also supports users to configure services, and service providers to manage resources according to system conditions.

REFERENCES:

1. Turumbetov, B.K., &Reypnazarov, E.N. (2019). Problems of creating ways to provide information security in e-mails. Science and Education in Karakalpakstan. Nukus, 9(1), 12-16.

2. Reypnazarov, E.N., &Lazarev, A.P. (2019). Developing a method for selecting the most optimal WAF. Science and Education in Karakalpakstan. Nukus, 11(3), 87-92.

3. Хужаматов, Х.Э., Рейпназаров, Е.Н., &Хасанов, Д.Т. (2020). Симсиз сенсор тармоқорқали телекоммуникация объектларинингшамол энергия таъминотиманбаларинимасофавий мониторинги. In “Иқтисодиётнингтармоқларининновационривожланишидаахборот-коммуникация технология-ларининиаҳамияти” Республика илмий-техник анжуманинингмаърузалартўплами, 5-6 март (pp.246-249).

4. Хужаматов, Х.Э., Хасанов, Д.Т., &Рейпназаров, Е.Н. (2020). Телекоммуникация объектлариэлектртаъминотиманбаларинимасофавий мониторинг тизиминиқурилиштамойиллари. In “Иқтисодиётнингтармоқларининновационривожланишидаахборот-коммуникация технология-ларининиаҳамияти” Республика илмий-техник анжуманинингмаърузалартўплами, 5-6 март (pp.249-251).

5. Siddikov, I.X., Hujamatov, X.E., Hasanov, D.T., Reypnazarov, E.N., &Toshtemirov, T.Q. (2020). Telekommunikatsiyatizimlariningavtomatikboshqariluvchiquyoshenergiyata’minotiniIoTmodellariasosidatadqiqetish. Muhammadal-Xorazmiyavlodlari. Tashkent, 12(2), 52-58.

6. Khujamatov, K.E., Reypnazarov, E.N., Khasanov, D.T., Nurullaev, E.E., &Sobirov, S.O. (2020). Evaluation of characteristics of wireless sensor networks with analytical

modeling. Bulletin of TUIT: Management and Communication Technologies. Tashkent, 3(2), p.8.

7. Сиддиков, И., Хужаматов, Х., Хасанов, Д., Рейпназаров, Е., &Ахмедов, Н. (2020). Телекоммуникация тизимларинингқуёш энергия таъминотиманбаларинимасофадан мониторинг қилишдастурийтаъминоти. Дастурийmaҳсулотучунгувоҳнома, DGU 09989, DGU 2020 2465.

8. Siddikov, I.K., Khujamatov, K.E., Khasanov, D.T., &Reypnazarov, E.N. (2020). IoT and intelligent wireless sensor network for remote monitoring systems of solar power stations. In World Conference Intelligent System for Industrial Automation WCIS 2020, 26-28 November (pp.186-195).

9. Khujamatov, K.E., Khasanov, D.T., Fayzullaev, B.A. &Reypnazarov, E.N. (2021). WSN-based research the monitoring systems for the solar power stations of telecommunication objects. IIUM Engineering Journal. Malaysia, 22(2), p.21.

10. Турумбетов, Б.К., &Рейпназаров, Е.Н. (2021). Расчет пропускной способности базовой станции сети LTE. In “Иқтисодиёттармоқларинингинновационривожланишидаахборот-коммуникация технологияларинингаҳамияти” Республика илмий-техник анжуманинингмаърузалартўплами, 4-5 март (pp.178-179).

11. Khujamatov, K.E., Reypnazarov, E.N., &Lazarev, A.P. (2020). Modern methods of testing and information security problems in IoT. Bulletin of TUIT: Management and Communication Technologies. Tashkent, 4(4), p.11.

12. Khujamatov, K., Khasanov, D., Reypnazarov, E., &Akhmedov, N. (2021). Existing Technologies and Solutions in 5G-Enabled IoT for Industrial Automation. Blockchain for 5G-Enabled IoT. Springer, Cham, 181-221.

13. Siddikov, I., Khasanov, D., Khujamatov, H., &Reypnazarov, E. (2021). Communication Architecture of Solar Energy Monitoring Systems for Telecommunication Objects. In International Conference on Information Science and Communications Technologies ICISCT 2021, 3-5 November (p.5).

14. Siddikov, I., Khujamatov, H., Reypnazarov, E., &Khasanov, D. (2021). CRN and 5G based IoT: Applications, Challenges and Opportunities. In International Conference on Information Science and Communications Technologies ICISCT 2021, 3-5 November (p.5).

15. Khujamatov, H., Siddikov, I., Reypnazarov, E., &Khasanov, D. (2021). Research of Probability-Time Characteristics of the Wireless Sensor Networks for Remote Monitoring Systems. In International Conference on Information Science and Communications Technologies ICISCT 2021, 3-5 November (p.6).

16. Khujamatov, H., Lazarev, A., Akhmedov, N., Reypnazarov, E., &Bekturdiev, A. (2021). Methods for Automatic Identification of Vehicles in the ITS System. In International Conference on Information Science and Communications Technologies ICISCT 2021, 3-5 November (p.5).

17. Xujamatov, X., Reypnazarov, E., Musaboyeva, N., &Aripov, J. (2022). Simsiz sensor tarmoqlariningishonchliliginitadqiqqilish dasturi. Dasturiymahsulotuchunguvohnoma, DGU 14860, DGU 2022 0479.

18. Reypnazarov, E.N. (2022). Analysis of the Frequency Allocation Experience in the World and Uzbekistan for 5G-Based Systems. International Journal of Innovative Engineering and Management Research. Institute for Advanced Studies, 11(3), 79-89.

19. Sudeep Tanwar,HalimKhujamatov,BairamTurumbetov,ErnazarReypnazarov and ZamiraAllamuratova,"Designing and Calculating Bandwidth of the LTE Network for Rural Areas," International Journal on Advanced Science, Engineering and Information Technology, vol. 12, no. 2, 2022. [Online]. Available: http://dx.doi.org/10.18517/ijaseit.12.2.14950.

20. Саттаров, Х.А., &Хужаматов, Х.Э. (2015). Методика развития конструкций датчиков угловых ускорений. Молодежь в науке: Новые аргументы. 204-207.

21. Khujamatov, Kh.E., Sattarov, Kh.A., Najmatdinov, K.M., &Anarbaev, M.A. (2016). Modeling and researching of the processes of control of hybrid power supply systems. In Материалы конференции “Информационные технологии и моделирование процессов в фундаментальных и прикладных исследованиях”, (pp.151-157).

22. Khujamatov, Kh.E. (2016). The quality of electrical energy in the three-phase electric networks. Проблемы и достижения современной науки. 1, 154-156.

23. Сиддиков, И.Х., &Хужаматов, Х.Э. (2016). Қайтатикланувчи энергия манбалариниўз ичига олган гибрид энергия таъминотитизимларинингбошқарувинимоделлаштиришватадқиқэтиш. “TATU xabarlari” ilmiy-texnikavaaxborot-tahliliyjurnali. 3, 60-66.

24. Хужаматов, Х.Э. (2016). Телекоммуникация объектларинибарқарорэлектрэнергиясибилантаъминлашдаавтономқуёшэлектрстанциясиниқўллаш. “TATU xabarlari” ilmiy-texnikavaaxborot-tahliliyjurnali. 4, 22-31.

25. TemirovA.A., OlimovaS.B. Smartuylarninggibridenergiyata'minotimanbalariningsignaluzatishdatchiklarivaaxborottiziminingfunksionalimkoniyatlari “Инновационғоялар, ишланмаларамалиётга: муаммоларваечимлар” Халқароилмий-амалий онлайн анжуман // Andijon 2020. 72-75.

26. Раджабов О.С., Темиров А.А., Соҳобиддинов А.А. Корхонатармоқларини “Ҳимояланган виртуал хусусийтармоқлар (VPN)” орқалиташкилэтиш. TATUxabarlari // 2019, №2 (49) 51-60 b.

27. TemirovA.A., SohobiddinovA.A., ToxirovX.O. Modellashtirishningasosiykontseptsiyalari, usullarivavositalari. “Машинасозлик” илмийхабарномаси 1 (13), 116-119.