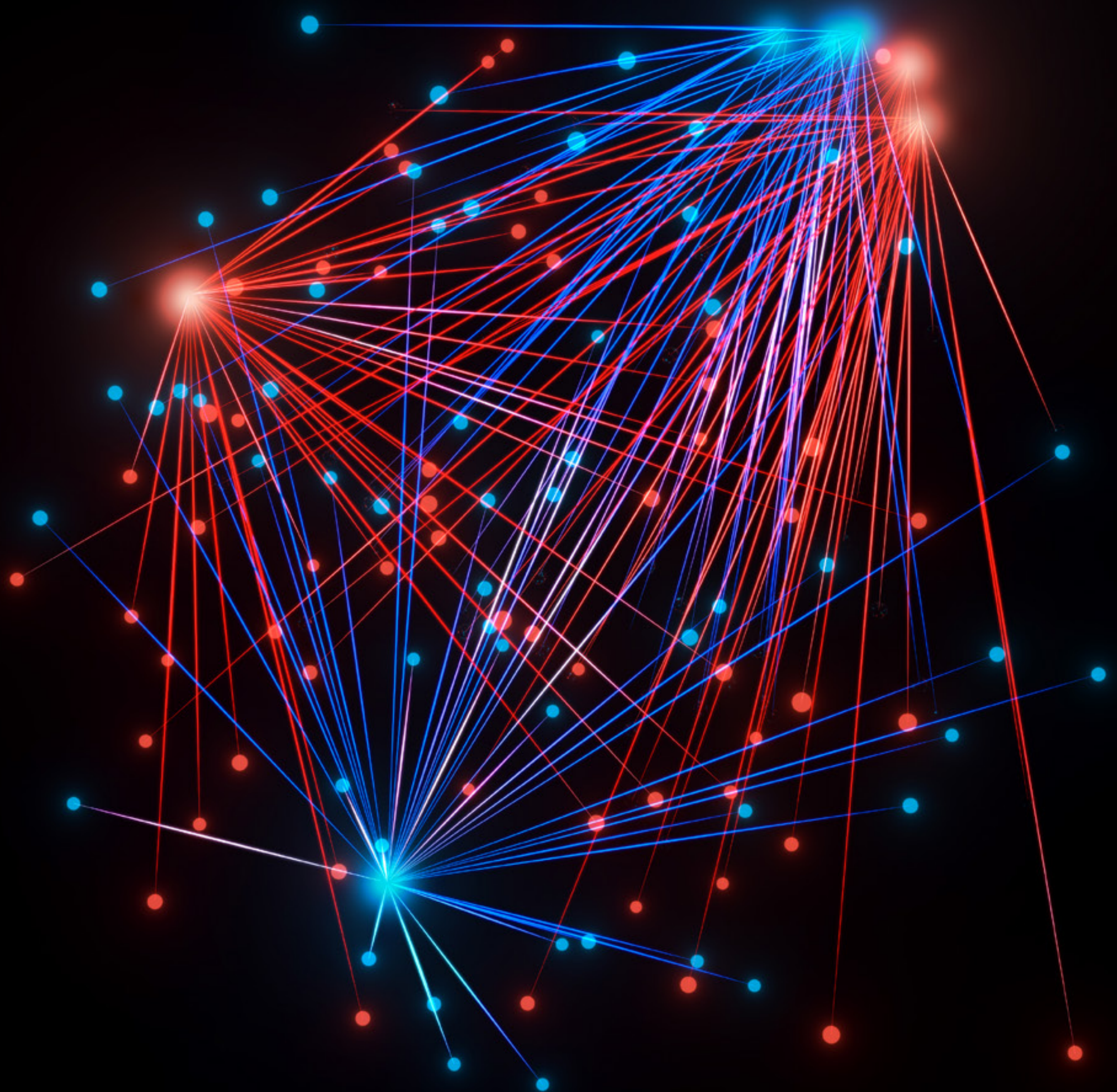
LTIMindtree



Deep Point of View

Performance Testing and Engineering for IoT Applications

Table of Contents

Introduction

It is fascinating that in today's world, machines and devices are having more interactions amongst themselves than humans at this point. This difference is set to widen as the Internet of Things (IoT) permeates our daily lives through smart appliances and smart cities.

An IoT application consists of IoT devices, the communication & computing infrastructure, and finally the end-user user interface. The IoT devices communicate with the communication infrastructure (aka gateway) through various protocols, which in turn communicate with the computing servers and then output is rendered on end-user's user interface via web and / or mobile applications.

According to 451Research, the total number of IoT devices around the world will surpass 14 billion by 2024. Organizations across the globe are determining what IoT means for their businesses and how to prepare for it. For example, System Integrators are working towards leveraging IoT devices for data extraction and analytics, and instantaneous control response in complex systems.

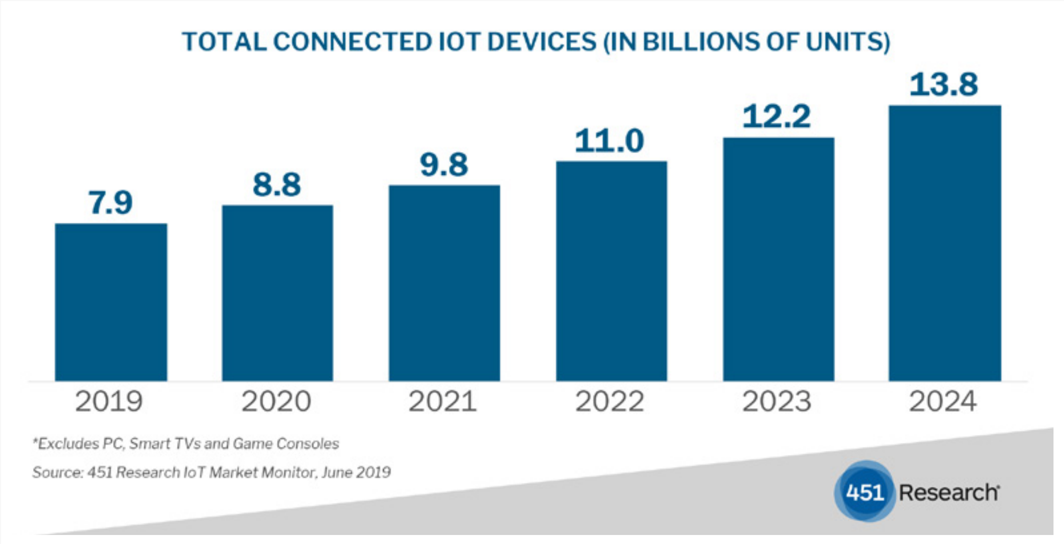


Fig. 1: Total Number of connected IoT Devices (*forecast) (Source: 451Research)

IoT applications are used across various industrial verticals for monitoring and controlling operations. In sectors like healthcare, IoT applications are used for critical use cases such as glucose-level monitoring, heart-rate monitoring, and alerting medical caregivers in case of breach of levels. In manufacturing, IoT sensors capture equipment data, analyze metrics, make predictions, discover anomalies, send alerts to prevent downtimes and failures, and safeguard machinery and the workforce.

One of the critical success factors for any IoT application is the performance metrics and how fast the application responds to high volume of date and with accuracy. Therefore, it becomes imperative that every IoT application undergoes the right Performance Testing including scalability, availability, load testing, responsiveness, and Performance Engineering practices. Personnel safety may be jeopardized if performance and scalability are compromised in IoT applications, especially in sectors like healthcare and manufacturing.

Performance Testing and Engineering helps to verify the speed/responsiveness of IoT applications and their scalability by sizing the necessary infrastructure capacity to meet the processing demands for handling large workloads. It also helps check the application's availability by determining whether the application's stability can be sustained for a longer duration under varying loads.

But, for performance testers and engineers, IoT presents significant change that may wind up disrupting long-standing testing techniques. An IoT ecosystem comprises multiple interconnected field devices of varied models and hardware consisting of IoT sensors, connectivity modules, smart devices, network, cloud-based applications, security systems, and data analytics systems. Creating a realistic test environment for such an ecosystem is almost impossible.

Also, there are no standard protocols available for IoT device communication. The IoT devices might use various non-standard protocols for communicating with the IoT gateway. Current Performance Testing and Engineering tools do not support all the IoT protocols. Thus, there is a need to have the right tooling strategy.

Below are some of the challenges that make the Performance Testing and Engineering of IoT applications different from traditional Web and Mobile Application Performance Testing.

- Diversity in IoT applications, devices, and platforms creates a nightmare scenario for testers.
- Lack of standard protocols available for IoT device communication.
- Fast-moving data and increased load straining the infrastructure.

In the following sections, we will present our understanding of this area and our approach to conducting Performance Testing and Engineering forIoT applications. We will start by summarizing the IoT ecosystem and some related industry and market trends.

IoT Ecosystem

Brief Introduction to the IoT Ecosystem

The Internet of Things (IoT) applications is a network of interconnected sensors and devices that collect and exchange data with various components in the different layers of the application infrastructure. They use various protocols to communicate with each other. Communication protocols used in IoT applications include UDP (User Datagram Protocol), MQTT (Message Queuing Telemetry Transport), CoAP (Constrained Application Protocol), AMQP (Advanced Message Queuing Protocol), and so on.

IoT application performance testing and engineering requires the simulation of realistic device workloads, network variations, and virtual gateways under real-life usage conditions to ensure the IoT ecosystem meets the expected performance and scalability characteristics. The IoT architecture has four main layers:

- **IoT Devices:** It includes the hardware devices that continuously send/receive messages to the IoT application through the IoT gateway.
- **Communication Infrastructure:** It supports the communication between the devices and the IoT Application. IoT gateway is the backbone of communication infrastructure. The IoT gateway records and stores data received from the thousands of field devices/sensors and forwards it to the IoT application.
- **Computing Infrastructure:** This is the middle layer that handles/analyses all the collected data. The computing infrastructure includes the message broker, BI and analytics, and other reporting servers that process the device data and provides meaningful, intelligent analysis insights. The Message Broker layer supports protocols such as HTTP and MQTT to digest the incoming device messages and passed on to the processing layer. Event Processing and Analytics takes the events from the broker, processes, acts upon it, and stores the data in the database.
- **End User UI Application:** This layer includes the web or mobile application where analytics results are made available to the end-users. This layer uses HTTP protocol to exchange data between the IoT application and other end-user applications. This layer consumes the data generated from the analytics application.

The following figure gives a view of all the 4 layers of an IoT Application Ecosystem and how the data flows to the end user mobile/web application (consumption layer) from the sensors and the IoT devices.

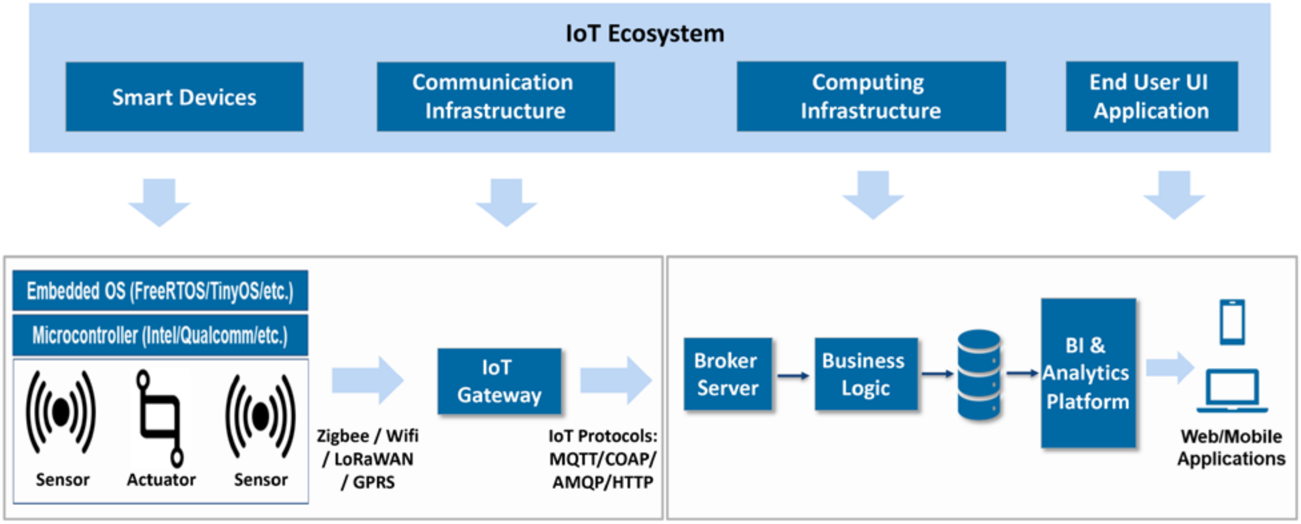


Fig. 2: IoT Ecosystem

IoT Reference Architecture

IoT architecture is made up of the following component layers:

- Device layer includes the various types of field devices that send/receive signals to/from IoT Gateway.
- Network layer supports the communication between the devices and the IoT Platform.
- Message Broker layer supports protocols such as HTTP and MQTT to digest the incoming device messages and passed on to the processing layer.
- Event Processing and Analytics layer takes the events from the broker, processes, acts upon it, and stores the data in the database.
- The Client Layer consumes the data generated from the analytics platform.

The below figure shows the reference architecture for IoT:

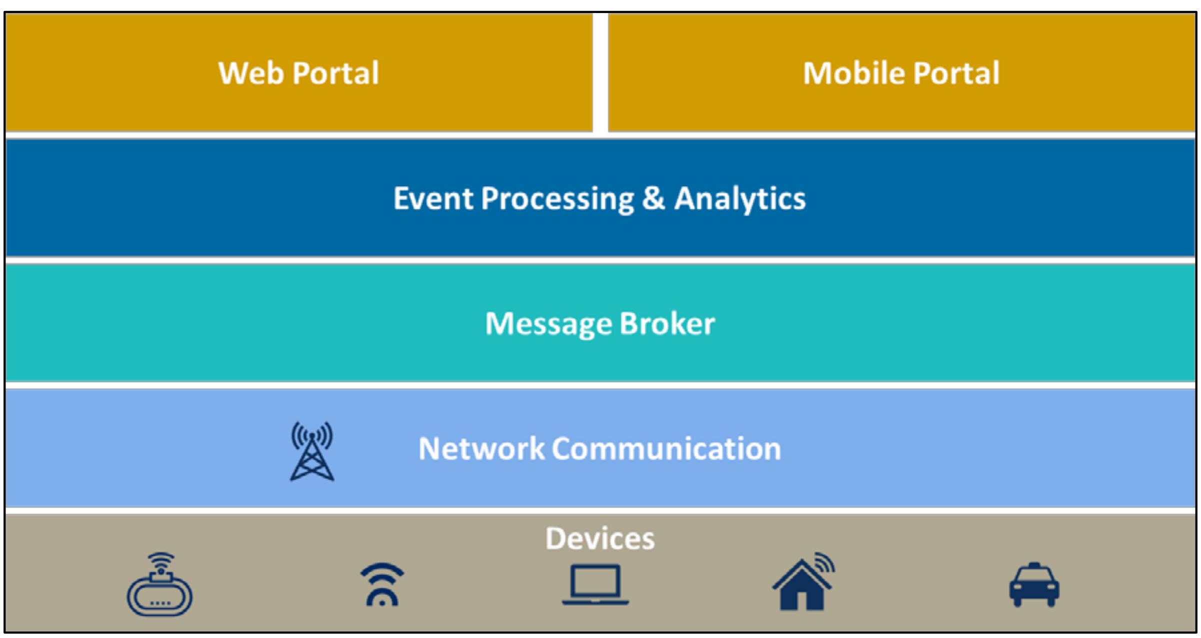


Fig. 3: IoT Reference Architecture

IoT Industry / Market Trends

According to McKinsey:

- **By 2030**, IoT could enable USD 5.5 trillion to USD 12.6 trillion in value globally, including the value captured by consumers and customers of IoT products and services.
- **Manufacturing** could generate around USD 1.4 trillion to USD 3.3 trillion in terms of value by 2030.
- **IoT's economic impact on healthcare** could reach between USD 0.5 trillion and USD 1.8 trillion by 2030.
- **Autonomous vehicles** make up the fastest-growing IoT-value cluster.

Source: McKinsey

Businesses are also incorporating IoT into their offerings, with a survey done in the World Quality Report stating that 97% of the respondents have an IoT presence in their products. With the increase in IoT product offerings, the market for end-user solutions is also expected to grow significantly.

Research firm MarketsandMarkets reported that the IoT solutions and services market will grow from USD 243.1 billion in 2022 to USD 575.0 billion by 2027, at a CAGR of 18.8% during the forecast period.

Need for Performance Testing and Engineering for IoT

Why do IoT applications require Performance Testing and Engineering?

With IoT devices and sensors increasingly connecting all physical objects and applications, there is a growing need to rethink the old performance testing and engineering methods for IoT applications. IoT is all about providing real-time data to users so they can make quick, educated decisions.

To guarantee that IoT applications can perform in real world scenarios, it is critical to conduct performance testing and engineering in the real world under less-than-ideal settings. This ensures the degree of quality regardless of what they are subjected to.

Below are some of the reasons stating the need for performance testing and engineering of IoT applications:

- Performance Testing and Engineering is critical for IoT applications as they validate the responsiveness/speed of the application.
- IoT Service layer performance and infrastructure capacity need to be validated for scalability by sizing the necessary infrastructure capacity to meet the processing demands for handling large workloads.
- It also helps check the IoT application's availability by determining whether the application's stability is sustained for a longer duration.
- For a successful IoT application, UI layer performance must be validated to ensure and enhance the end-user experience.
- Bringing early performance engineering will help deliver a high-performing, scalable, and resilient IoT application ecosystem that meets the high availability SLAs.
- Network Communication forms an important aspect of IoT applications. The performance impact due to real-time network conditions between devices and the IoT Gateway needs to be validated proactively before impacting the end-user experience.
- IoT projects involve risks concerning performance and availability.

LTIMindtree

Key Areas under Performance Testing and Engineering for IoT Applications

As per the IoT application architecture, there are 3 key areas for Performance Testing and Engineering.

01
Device to Device (D2D) Communication (Peripheral device to peripheral gateway)

02
Device to Server (D2S) Communication (Peripheral gateway to Server)

03
Server to Server (S2S) Communication (MQTT Broker to BI server)

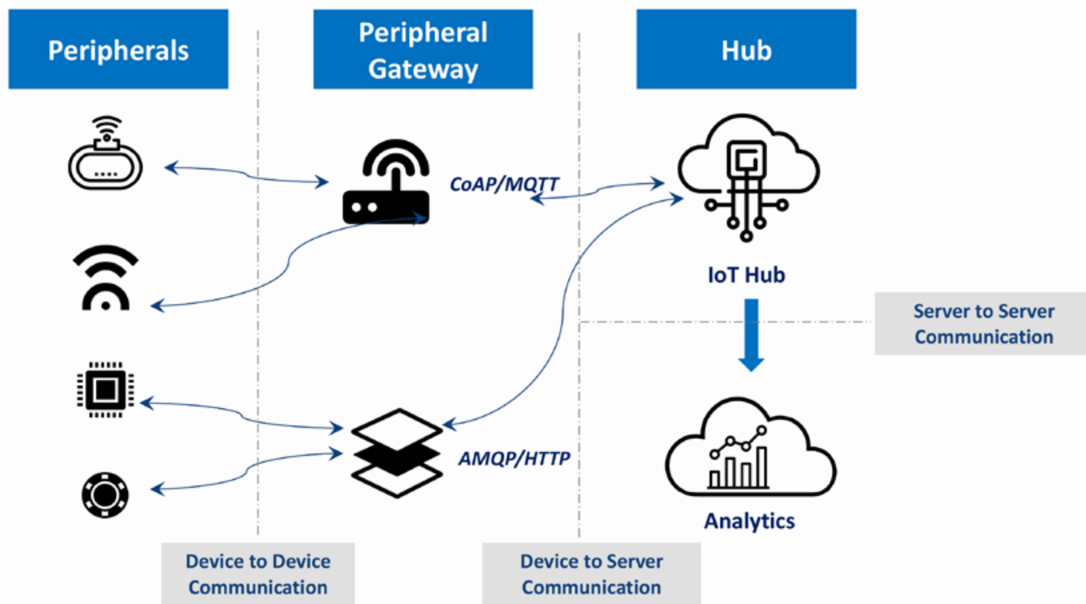


Fig. 4: IoT Application Architecture

IoT application Performance Testing and Engineering includes performance analysis of the following components:

1. Device Platform Layer Performance Analysis

- The device layer doesn't exist in traditional web applications. Device diversity and using non-standard protocols for device-to-application communication is the major challenge in creating realistic workloads.
- Device layer sends and receives messages (bi-directional) continuously (24 x 7) on a real-time basis.
- Performance Benchmarking on various types of actual devices will be challenging. There could be the use of low-power device platforms like Arduino, Raspberry Pi, etc.
- The devices use low-power communication technologies like Zigbee, LoRaWan, etc. Network simulation with packet losses and latency issues might be required for realistic workload creation. Device workloads need to be simulated with the network variations (Bluetooth/WIFI/2G/3G/4G/5G).
- The device communicates with the IoT Gateway through lightweight communication protocols like MQTT, CoAP, AMQP, XMPP, etc., apart from HTTP-based REST API communication.

2. **Performance Benchmarking of IoT Cloud Service Providers**
    - Use of Azure IoT Hub / AWS IoT Core services / Cisco IoT Cloud Connect / etc. to benchmark and analyze the pros and cons of various service providers.

3. **IoT Gateway Performance and Capacity Analysis**
    - Performance benchmarking and sizing of IoT gateway to handle target message volumes per unit time.

4. **Performance Benchmarking of Message broker servers**
    - (Example: ActiveMQ /VerneMQ / Amazon SQS / RabbitMQ / Kafka /Mosquitto / etc.) - Identify the maximum supported load level and the breakpoint of the MQTT Broker.

5. **Performance Analysis of Device simulation solutions to generate device data**
    - Performance simulation of extensive device data (Test data simulation for continuous, prolonged duration to create real-time workload).
    - Exploration of various IoT device simulation platforms required.
    - Analyze the performance implications and best practices for using various device simulation solutions like IoTIFY, IBM Watson, MATLAB, etc.

6. **Performance Benchmarking of IoT platform** for handling a sudden spike in large burst device loads.

7. **Validation and optimization** of the IoT platform performance for realistic workloads by simulating message flows at scale; a mass disconnect and reconnect of connected assets; bi-directional platform-initiated activities with bulk SOTA or FOTA updates.

8. **Simulators / Emulators to generate realistic message volumes** (As in real devices) for the creation of continuous workload for a prolonged duration to create real-time workload. The communication between devices and gateway needs to be studied and simulated.

LTIMindtree

Challenges in IoT Application Performance Testing and Engineering

As per our understanding, the following are the challenges faced during Performance Testing and Engineering of IoT applications:

- Need for Performance Testing and Engineering Tools that support IoT protocols.
- Need for tools for replicating device usage patterns and continuous workload levels.
- Need for infrastructure to generate high load tests to simulate load from millions of devices.
- Measuring local processing power and other performance KPIs on various devices considering diversity in the device types and models.
- Need for Network emulation tools (3G/4G/Bluetooth/Wi-Fi/ZigBee/LoRaWAN etc.) to perform realistic network simulation across devices.
- Need for sophisticated observability tools (Application Performance Management tools) to analyze the application health and infrastructure health of various server components of IoT architecture.
- Need for effective Test data management strategy for creating large synthetic data volumes.
- Need for investment for setting up IoT Test lab environment.

To overcome these challenges, below are the key points that need to be looked at

- The architecture and the communication mechanism should be thoroughly analyzed before the right tool with the supported protocol can be finalized.
- IoT application Performance Testing and Engineering requires massive load generation and workload simulation, and this needs to be planned perfectly well in advance.
- As the IoT environment is vast and complex, everything cannot be simulated in the environment. A tradeoff needs to be made between different types of devices and networks.
- Network emulation is required for thorough IoT Performance Testing and Engineering, and it needs to be planned early.
- For end-to-end monitoring of the devices, industry-standard observability tools can be procured.
- Observability tools should be integrated with IoT Performance Testing and Engineering.

Specialized IoT Performance Testing and Engineering Tools do exist that can support multiple protocols, emulate networks, and support cloud-based load generation. A lab needs to be set up which can be a place for executing the various types of testing, to do R&D work, for training and skilling the talent in this area, and for building use cases with the right ecosystem of tools and infrastructure built in.

Performance Testing and Engineering for IoT Applications

IoT Application Performance Testing and Engineering Methodology

The Performance Testing and Engineering methodology for IoT applications is based on the regular performance tests that we carry out for traditional applications. But as the overall usage pattern and behaviour of IoT applications is different from traditional applications, there is variation is their testing methodology. Below are the key steps in the methodology.

1. **Identification of the Testing environment and the right testing tool:** The IoT application ecosystem and protocols used for communication should be thoroughly analyzed. Based on this analysis, a Performance Testing & Engineering tool that supports the required IoT protocol (like MQTT, AMQP, CoAP, HTTP etc.) should be identified.

2. **Identification of the performance metrics:** Metrics based on the unique characteristics of IoT applications should be identified. Some of them are response time, network connectivity and packet loss, message success and failure etc.

3. **Planning and designing the tests:** Tests should be designed to create a geographical distribution of workload with varied network conditions. The performance & scalability characteristics of the IoT platform need to be validated for the realistic workload condition.

4. **Configuring the test environment:** All the necessary testing and monitoring tools should be arranged in order to prepare the testing environment.

5. **Creating the test scripts:** The test scripts should be developed as per the IoT protocol (supported by the Performance Testing & Engineering tool).

6. **Test Execution:** The tests should be executed. The test data that is generated should be captured and monitored.

7. **Analysis:** Application Performance Management (APM) tools can be used to monitor & diagnose the infrastructure & application health issues and optimize the IoT application performance to meet the desired SLAs. LTIMindtree has its own in-house observability tool called LTIMindtree Canvas Resilience which can be used to monitor the applications. It is a Unified Resilience Engineering platform built for hardening applications to deliver business resilience via chaos engineering, continuous observability, proactive resilience insights and SRE dashboards.

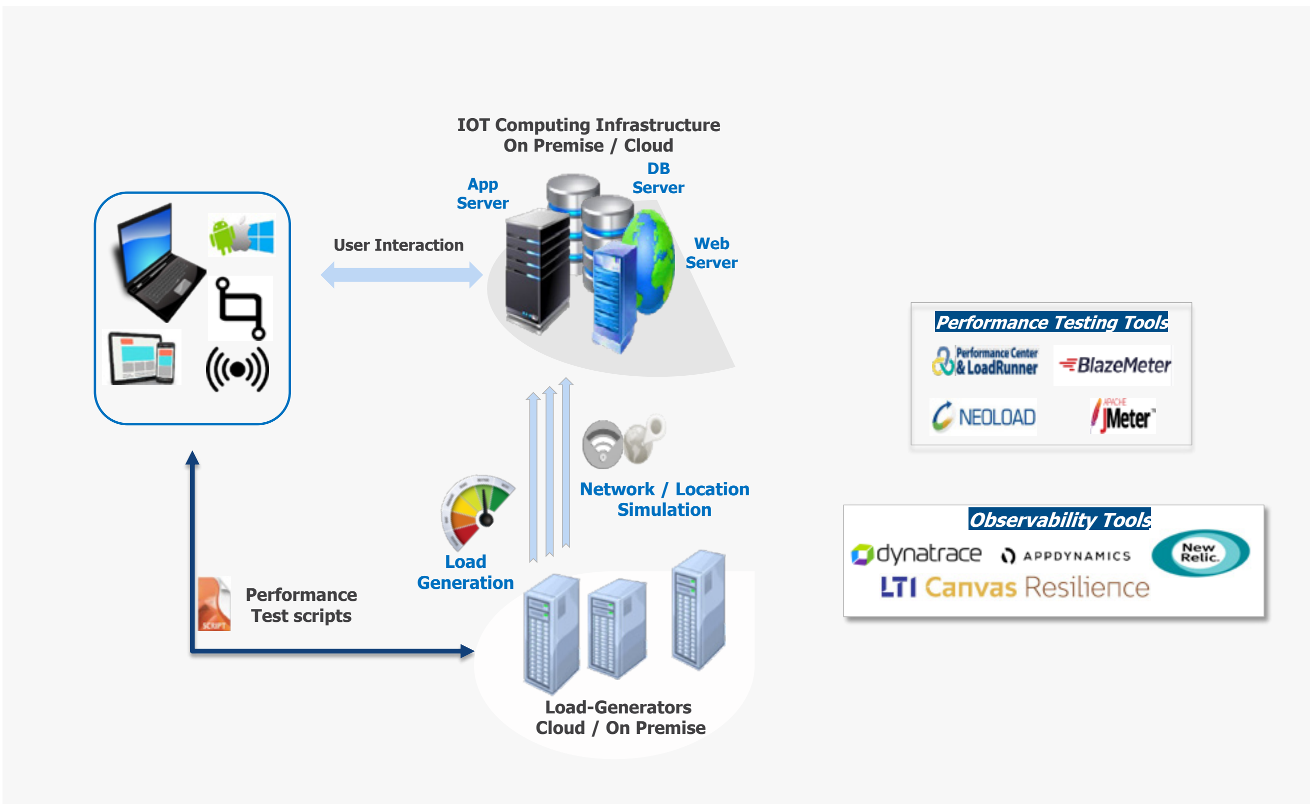


Fig. 5: IoT Application Performance Testing & Engineering Methodology

Performance Testing and Engineering Methodology for MQTT Protocol testing using the JMeter tool

Below we have defined the Performance Testing and Engineering methodology for the MQTT protocol using the open-source Apache JMeter tool. The MQTT protocol available in JMeter is used to generate the distributed load both from on-prem and cloud-based load generators to create a performance benchmark and identify the breakpoint of the MQTT broker. MQTT is a publisher and subscriber based protocol for low signal messages. Any client can request the broker for a message published by it and the broker decides based on the subscription model if the message can be delivered to the client.

As we can see in the figure below, we have the Application Under Test with the MQTT broker-based subscription. For load generation using JMeter, it can be done on-premises which could be based on master-slave architecture or through the cloud.

As a typical IoT application could have millions of devices, simulation of load on-premises would be very complex, time consuming, and involve high costs. This would not be the case for traditional applications, where load only needs to be generated for a few thousand users. Hence for IoT applications, cloud based load generation is recommended.

Full stack observability is also required to assess the behaviour of the application under varying network conditions and sudden increase in the load. Canvas Resilience tool can be used for full stack observability.

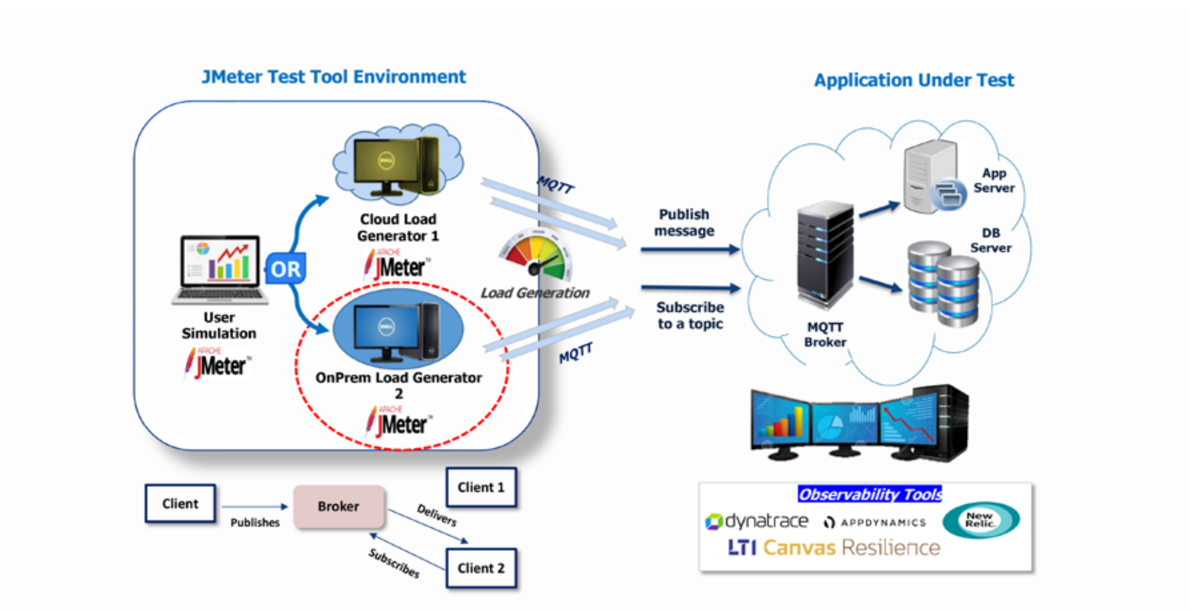


Fig. 6: Performance Testing & Engineering for MQTT Protocol using JMeter'

IoT Performance Testing and Monitoring Tools

- Many of the popular Performance Testing and Engineering tools like Microfocus LoadRunner, open-source Apache JMeter, Tricentis NeoLoad, etc., support IoT protocols like MQTT, AMQP, and CoAP apart from the HTTP protocol.
- Performance Testing and Engineering tool, which includes an SDK to code the required protocol, is required as there are several non-standard protocols used in IoT applications that might not be supported by the Performance Testing and Engineering tool.
- Specialized IoT Performance Testing and Engineering tools exist, such as MachNation Tempest, which supports MQTT, WebSocket, Modbus and LoRaWAN, and custom protocols.
- Many device simulation solutions like IoTIFY, MATLAB, and IBM Watson can be used to simulate the device workload to meet the varied configurational and network requirements.
- Cloud-based load generation across geographical locations with network variation simulation is facilitated by most of the above Performance Testing and Engineering tools.
- Service virtualization tools required to mock the notifications sent/received from devices/sensors is facilitated by most of the above Performance Testing and Engineering tools.
- Network simulation is an important element to be considered during Performance Testing and Engineering. As the devices and users connect to the IoT platform in constrained network connections, the ability to simulate packet loss and introduce network latency bandwidth constraints is essential. Network simulation capabilities are supported by most of the commercial Performance Testing and Engineering tools.
- Comprehensive monitoring of IoT applications using Application Performance Management (APM) tools like AppDynamics, Dynatrace, New Relic, etc. is essential to analyze the performance impact of various components. If the application is hosted on the cloud, then cloud-native performance monitoring using AWS CloudWatch or Azure Monitor will also help in monitoring, diagnosis and troubleshooting performance issues.

The below table shows the popular IoT Performance Testing and Engineering tools.

Sl. No	Tool	Protocols Supported
1	Apache JMeter	HTTP, CoAP, AMQP, MQTT and Kafka
2	Microfocus LoadRunner	MQTT, CoAP, HTTP
3	SmartBear Load UI	MQTT, AMQP, HTTP
4	RabbitMQ Perf Test Tool	MQTT, AMQP, HTTP, WebSocket
5	Tricentis NeoLoad	MQTT, HTTP, AMQP
6	Locust	MQTT
7	MachNation Tempest	MQTT, HTTP(s), WebSocket, Modbus, LoRAWAN and Custom Protocols
8	MATLAB	REST, MQTT, and OPC UA
9	IBM Watson IoT Simulator	MQTT and HTTP REST
10	IoTIFY	MQTT, HTTP, CoAP, AMQP, LWM2M and Custom Protocols

Table 1: Popular IoT Performance Testing & Engineering Tools (Multiples Sources)

Performance Testing and Engineering KPIs

Below performance KPIs are reported by simulating realistic workload conditions by bringing in Network variations and packet loss/latency conditions.

- Response time from device to IoT platform (server) communication and IoT platform (server) to device communication to carry out a specific action.
- Messages per second handled by MQTT Broker.
- Number of device connections to IoT platform per MQTT broker instance.
- Maximum latency by time.
- Message Success and Failure rate.

- Messages per second handled by IoT gateway.
- Broker Server Infrastructure requirements to handle target data volumes (CPU / Memory / Disk / Network utilization details).
- Big data processing speed and server capacity requirements to handle target data volumes (CPU / Memory / Disk / Network utilization details).
- Application response time (web or mobile UI) for the expected concurrent user loads.

IoT Application Performance Testing and Engineering vs. Traditional Web Application Performance Testing

IoT Application Performance Testing and Engineering is different from traditional web application testing in below ways.

PARAMETERS	WEB APPLICATION	IOT APPLICATION
Communication Protocol	HTTP / HTTPS, FTP, SMTP, JDBC, etc.	MQTT, CoAP, AMQP, UDP, XMPP, HTTP, etc.
Types of Workloads considered	Usually, the user-initiated workload is considered.	Device-initiated and IoT platform-initiated workloads are considered.
Simulation	End user behaviour simulation.	Devices and Sensors behaviour simulation using MATLAB, IoTIFY, IBM Bluemix (now IBM Cloud), etc.
Performance Test Coverage	Application Layer.	Physical, Network and Transport layers.
Big Data and BI	Not all web applications require performance analysis of Big Data processing and BI analytics.	Performance analysis of huge volumes of data from IoT devices are analyzed using Big Data, BI and analytics solutions for intelligent decision making in all IoT applications.
Requests / Responses	Users create the requests and receive the responses.	IoT devices create requests and receive responses as well as requests, and they also provide responses.

Table 2: Difference between Web Application and IoT Application Testing

Due to the above differences the Performance Testing and Engineering of IoT applications are impacted in the below ways:

- Devices in IoT applications communicate with each other through multiple protocols/communication mechanisms, so the appropriate Performance Testing and Engineering tool needs to be used which supports the given protocol/communication mechanism.
- As the IoT ecosystem consists of numerous different kinds of devices, creating a realistic test environment and hence realistic test scenarios is very hard.
- Unlike traditional web application testing, where the traffic can be recorded from the web browser or the application, recording traffic from the IoT devices and gateway is not straightforward.
- IoT devices and sensors on endpoints are sometimes connected to restricted networks, leading to packet losses and latency issues. Hence before creating scenarios, the workflow needs to be clearly understood.

LTIMindtree

Verticals and their Use-cases

Companies have adopted IoT technology across industries to deliver new capabilities to their customers. A lot of companies have an IoT presence on their products. As performance and scalability are important characteristics of the IoT ecosystem that need to be validated, IoT application Performance Testing and Engineering would be relevant for these industries.

Manufacturing

- **Asset/Plant performance optimization** – Businesses use IoT technology for capturing data, visualizing it, and analyzing it to improve the reliability and availability of physical plant-floor assets. Performance Testing and Engineering of these IoT applications, which are relied upon for the maintenance and optimization of plant assets, would be of importance as the performance of these assets is affected by it.
- **Remote Asset monitoring and control** – Companies are adopting remote monitoring and control IoT-based solutions, which lets their engineers and service teams remotely monitor and control the assets in manufacturing plants and hence optimize production. Validating such solutions' performance would help increase worker safety, productivity, and asset maintenance.
- **Predictive Maintenance** – Businesses use Predictive Maintenance solutions to forecast the remaining useful life of assets and ensure they are repaired before they fail. Such solutions have enabled businesses to actively monitor their critical assets and avoid downtime. Validating the response times of the IoT devices monitoring the assets would help in the effective detection of any faults and hence improve asset performance and reliability.
- **Energy Optimization** – IoT sensors can detect energy use and automatically adjust industrial systems' operations and HVAC systems to use as little electricity as possible.

Healthcare

- **Remote Monitoring and Patient Care** – Healthcare providers are using IoT to offer remote monitoring and care. It helps providers remotely assess patient's health metrics, such as glucose levels, heart rate, blood pressure, and body temperature.
- **Sending alerts** – Medicine manufacturers have installed chips on their prescription bottles to send alerts reminding patients to take their dose. These IoT devices need to send on-time alerts, as the patient's health is affected by it.
- **Tracking Staff, Patients, and Inventory** – IoT devices help hospitals and medical facilities facilitate asset tracking and monitoring of day-to-day activities in the facility.

Transportation and Logistics

- **Fleet Management** – IoT helps automate fleet management processes, collecting and analyzing data from the fleet and send it to the cloud. This helps analyze the fleet's performance and allows for predictive maintenance.
- **Location Tracking** – Logistics companies use location-based IoT technologies to track shipments and reroute drivers when obstacles arise. Knowing where the assets are at any time promotes efficiency.

Government

- **Smart Cities** – Cities around the world are optimizing their infrastructure for efficiency and sustainability through smart city development and IoT. The data collected by the sensors provide insights into repair needs, waste management, and energy efficiency. The sensor data also helps glean insights into public space surveillance, road and traffic management, and parking management.
- **Disaster Prevention** – IoT technology helps monitor temperature, water levels, or seismic activity to combat forest fires, protect land, safeguard power plants, and plan.

Energy

- **Solar/wind monitoring –** IoT devices and sensors can be used to monitor solar and wind energy generation. They can help detect any failure or abnormal decrease in energy efficiency.
- **Backup power monitoring** – IoT devices are used to monitor backup power levels. This helps to prevent complete service breakdown due to failure of battery back-up.

Conclusion

The growing investment in IoT across sectors is expected to increase the demand for testing services. Mordor Intelligence opines that the IoT testing market, which was valued at USD 1.1 billion in 2020, is expected to reach USD 6 billion by 2026 at a CAGR of 32.34%. This would foster the demand for managed services. As per Gartner, the IT Services for IoT market will represent a 58-billion-dollar opportunity in 2025.

Performance Testing and Engineering of IoT applications is complex and very different from traditional web application performance testing. Many challenges are involved, but they can be overcome with the right strategy. As IoT devices are seeing an increase in application across various industries, there is a high demand for assessing and certifying IoT applications for their performance, scalability, and resilience characteristics.

For the service providers, IoT is a significant source of business opportunities. Not only are managed IoT testing services expected to witness significant growth, but the Performance Testing and Engineering services market across all three IoT layers is promising.

Fig. 7: IoT Testing Market CAGR Growth

Authors

Manish Potdar
Head – Incubation & Industrialization

Sachin Jain
Head – Crystal and Deep POV

Rex Jesudas P
Head – Industry 4.0 COE

Govindan Ramasamy
Head – Performance Engineering Services

Bharat Trivedi
Principal – Enterprise Architecture

Ramya RamalingaMoorthy
Industrialization Head – Performance Engineering Services

ArunKumar Dutta
Lead Architect – Performance Engineering Services

Mangesh Ghangale
Lead – IoT Testing Services

Chitrang Negi
Research Analyst – Crystal and Deep POV

Hakimuddin Bawangaonwala
Research Analyst – Crystal and Deep POV

LTIMindtree

References

1. Forecast: IT Services for IoT, Worldwide, 2019-2025 (gartner.com)
2. IoT Solutions and Services Market Size, Share and Global Market Forecast to 2027 | MarketsandMarkets
3. Where and how to capture accelerating IoT value | McKinsey
4. Internet of Things (IoT) Performance Testing and Engineering - Load Testing Software and Web Load Testing Solutions from Radview
5. IoT Testing Market Share, Growth (2022 - 27) | Industry Outlook (mordorintelligence.com)
6. IOT Performance Testing and Engineering and its Challenges – QAElite Souls
7. Tech Mahindra Accelerates into IoT Performance Testing and Engineering for Connected Cars - NelsonHall (nelson-hall.com)
8. IoT Testing and Validation Services - Offerings | Infosys
9. IoT Testing | Automation Testing Services - Wipro
10. IoT Testing (wipro.com)
11. Tools and their supported protocols
    a. LoadRunner – LoadRunner Professional and LoadRunner Enterprise Supported Protocols (microfocus.com)
    b. SmartBear – About IoT MQTT Testing | ReadyAPI Documentation (smartbear.com)
    c. RabbitMQ – Which protocols does RabbitMQ support? — RabbitMQ
    d. Neoload – IoT Shared Application - Neotys
    e. IOTIFY – Introduction - IoTIFY Network Simulator
    f. MachNation Tempest – Home - TEMPEST by MachNation
    g. Locust – Performance Testing and Engineering of IoT Applications: PerfCast - Summer 2018 (qasource.com)
    h. MATLAB – Internet of Things - MATLAB and Simulink (mathworks.com)
    i. IBM Watson IOT Simulator – IBM Watson IoT Platform (ibmcloud.com) / Simulating device data | IBM Cloud Docs
12. Performance Testing and Engineering of IoT Applications: PerfCast - Summer 2018 (qasource.com)
13. The top 10 IoT Use Cases (iot-analytics.com)
14. 6 Exciting IoT Applications in Healthcare (iotforall.com)
15. Industry-Leading IoT Use Cases 2022 | Datamation
16. Use cases for the Internet of Things Across Industries (twilio.com)
17. Top 5 industrial IoT use cases (techtarget.com)
18. Performance testing 101: How to approach the Internet of Things - Tricentis
19. How Effective Performance Testing Can Help Build Robust IoT Systems (einfochips.com

LTIMindtree

Appendix

List of Figures

List of Tables

LTIMindtree is a global technology consulting and digital solutions company that enables enterprises across industries to reimagine business models, accelerate innovation, and maximize growth by harnessing digital technologies. As a digital transformation partner to more than 700+ clients, LTIMindtree brings extensive domain and technology expertise to help drive superior competitive differentiation, customer experiences, and business outcomes in a converging world. Powered by nearly 90,000 talented and entrepreneurial professionals across more than 30 countries, LTIMindtree — a Larsen & Toubro Group company — combines the industry-acclaimed strengths of erstwhile Larsen and Toubro Infotech and Mindtree in solving the most complex business challenges and delivering transformation at scale. For more information, please visit **www.ltimindtree.com.**