

IoT Testing - The Big Challenge Why, What and How

Benny Sand^(✉)

Jerusalem, Israel
benny.sand@gmail.com

Abstract. “Internet of Things” (IoT) is the network or associations between those Internets connected objects (smart Devices) that are able to exchange information by using an agreed method and data schema. The enormous amount of things, the communication protocols the combination between hardware and software, the Big Data impact, a verity of protocols, lack of standards and the high level of the required SLA by the end users set up new and challenging bars regarding QA in general and testing in particular in the IoT scene. This article will highlight the challenges as well as address potential strategies and solutions.

Keywords: IoT testing · Testing assurance · DevOps interoperability · Security · Validation · Protocols

1 Introduction

The Internet of Things (IoT) is a key enabling technology for digital and virtual technologies Apparently 3.5 Billion things were connected in 2014, and the figure is expected to rise to 25 Billion things by 2020.

The recent progress on Internet of Things deployments with the rise of Mobile culture have given a strong push to the IoT to be today’s considered as one of the most promising emerging technologies. However the conceptual realization of Internet of Things is far from achieving a full deployment of converged IoT services and technology.

One of the key elements in the IoT go to market path is Interoperability. Interoperability can be generalized as the feature for providing seamless exchange of information to, for example, personalize services automatically or simply exchanging information in a way that other systems can use it for improving performance, enable and create services, control operations and information processing.

2 IoT Challenges

Internet of Things enables the things/objects in our environment to be active participants, i.e., they share information with other objects and/or communicate over the networks (wired/wireless) often using the Internet Protocol (IP). Processing the IoT data enables to recognize events and changes in the surrounding environments and “things” can act and react autonomously. However, all these require heterogeneous

objects to exchange information in an interoperable way to make their data and services accessible and interpretable by other objects and services.

The IoT is an emerging area that not only requires development of infrastructure and technologies but also deployment of new services capable of supporting multiple, scalable (cloud-based) and interoperable (multi-domain) applications in a verity of telecommunication protocols. The significant IoT problem to be challenged is the interoperability of the information and services.

IoT refers to objects (“things”) and the virtual representations of these objects on the Internet. It defines how the things will be connected through the Internet and how those things “talk” amongst other things and communicate with other systems in order to expose their capabilities and functionalities “services”.

IoT is not only linking connected devices by using the Internet; it is also web-enabled data exchange in order to enable systems with more capacities to become “smart”. In other words, IoT aims to integrate the physical world with the virtual world by using the Internet as the medium to communicate and exchange information.

IoT is mainly supported by continuous progress in wireless sensors and actuator diversified networks and by manufacturing low cost and energy efficient hardware for sensor and device communications. However, heterogeneity of underlying devices and communication technologies and interoperability in different layers, from communication and seamless integration of devices to interoperability of data generated by the IoT resources, is a challenge for expanding generic IoT solutions to a global scale.

Networking everyday objects to send and receive data has been received with as much hope and promise as it has worry and concern. Certainly, the day may come when your refrigerator automatically orders milk when you are running low, but a connected supply chain might just as likely be shut down by a security breach by malicious hackers.

3 IoT Testing Challenges and Vision

Software testing helps in finalizing the software application or service against business and user requirements. It is very important to have good test coverage in order to test the software application completely and make it sure that it’s performing well and as per the specifications. While determining the coverage the test cases should be designed well with maximum possibilities of finding the errors or bugs.

Today’s connected world unifies multiple company aspects, namely customer engagement channels, supply chains, interfacing devices and application touch points. Therefore Quality Assurance organizations need to assess their customer experience capabilities, as well as ensure the functionality of each individual application; introducing remarkable developments in quality, cost and agility.

Companies need to focus on the disruptive nature of digital technologies by paying close attention to customer experience-based testing. The key to successfully executing this new approach is to look for service offerings that feature an integrated test delivery platform, encompass omni-channel test automation frameworks, mobile testing strategies and crowd testing.

With the brilliance of a connected world, comes the necessary capability to provide more niche expertise closer to the customer and the realization that testing is a combination crowd testing in order to reflect real life conditions to ensure a delivery of top notch IoT services.

In a connected world, global companies realize that they need to organize their Quality Assurance and testing functions with a combination of centralized and decentralized approaches. A testing team tightly integrated into the product development process is vital for complex integrations and transformations pr in other words Agile will become the governing model and will implemented via DevOps platform. Moreover, companies need to ensure they define their own formula for success as one size does not fit all. It is vital that they look for a testing partner with a multi-layered test target operating approach, continuous delivery integration and outcome and output-based pricing models; all governed by a 24/7 real-time dashboard.

Companies need to stop the one-way upstream integration and align it with a downstream approach to create a new TestOps concept. In order to stay ahead of the game, companies need to drive efficiency through risk-based analysis techniques, risk-based testing, test-driven development, integrated test delivery, and service virtualization.

Understanding that security and performance testing is a top priority area; companies need to include multichannel and behavior driven testing models and approaches as well as focused platform migration testing. It is crucial to have strong links with test automation framework, connected world test strategies, end user performance analysis, and competition benchmarking capabilities.

In a connected world, it is vital for applications to be tested on numerous operating systems and devices in different geographies; such ample testing cannot be done on premise, it must be done in the cloud. This is why it is important to ensure your testing partner has access to the best possible testing environments that leverage all necessary services.

In a connected world, competition is rapidly increasing, so companies need to closely examine these trends and ensure they are following the right steps to enrich their test methodologies. Implementing the right testing practices will allow companies to seamlessly manage the complexity and scale that IoT presents.

4 The Interoperability Impact in IoT

4.1 The Interoperability ExFactor

Interoperability is a major theme in the IoT scene; hence it impacts the testing lifecycle of Internet of things strategic and operational wise. Interoperability in IoT is compound as well as influenced form several elements which impact in a direct and indirect way on the implementation process.

The Technical Interoperability is usually associated with hardware/software components, systems and platforms that enable machine-to-machine communication to take place. This kind of interoperability is often centered on verity of communication protocols.

The Organizational Interoperability, as the name implies, is the ability of organizations to effectively communicate and transfer (meaningful) data (information) even though they may be using a variety of different information systems over widely different infrastructures, possibly across different geographic regions and cultures. Organizational interoperability depends on successful technical, syntactical and semantic interoperability.

Needless to say those two things cannot interoperate if they do not implement the same set of services. Therefore when specifications are including a broad range of options, this aspect could lead to serious interoperability challenges. Solutions to overcome these aspects consist of definition clearly in clear requirements the full list options with all conditions. In the latter case, defining profile would help to truly check interoperability between two products in the same family or from different family if the feature checked belongs to the two groups.

4.2 Methodologies for Interoperability Testing in IoT

Interoperability testing involves testing whether a given software program or technology is compatible with others and promotes cross-use functionality. This kind of testing is now important as many different kinds of technology are being built into architectures made up of many diverse parts, where seamless operation is critical for developing a user base.

The factors in interoperability testing include syntax and data format compatibility, sufficient physical and logical connection methods, and ease of use features. Software programs need to be able to route data back and forth without causing operational issues, losing data, or otherwise losing functionality. In order to facilitate this, each software component needs to recognize incoming data from other programs, handle the stresses of its role in architecture, and provide accessible, useful results.

Interoperability testing can be addressed in two main approaches for testing:

The empiric approach of testing regroups several ways to do testing. Since this kind of testing is informal, they are generally carried out while coding. There is no set procedure for informal testing, and this is entirely up to the coder to implement without the need to submit the test reports. The coder feels confident that his code works as required and contains no obvious bugs.

Empiric approach for testing encompasses tests that are done while developing the product to identify bugs, as well as those that is done on the fly.

The main advantages of the empiric testing methodology are the following: Tests can be done very earlier while developing the products, allowing detecting errors/bugs in the earlier stage of the development, moreover the tests can be setup very quickly, without huge constraints such as having reports to prepare etc.

Having said that the above e advantages can be canceled by the following drawbacks:

No ideas regarding the test coverage since there is no real test plan, part of the properties to be tested cannot be measured. Thus, errors/bugs may not be detected.

Since these tests have been done informally, end users will have difficulty to trust the final product so the marketing and business demerger can be rather significant. The methodological approach for testing generally encompasses different steps leading to the execution step where test suites are generated against products. These products can be at different degree of their development.

Three main steps can be seen in this approach: Abstract Test Suite (ATS) specification, Derivation of executable test, and Test execution and results' analysis.

The advantages of the methodological approach are the following: Improved test coverage due to a real consist methodology that monitors the whole processes while maintaining KPI's, properties to be tested can be measured. Thus, it may help in determining more precisely how to cover important parts of the system and sub-systems under test. By this way, it may reduce non-interoperability of the product at the end.

Moreover the methodological approach provides real added value to the market. As these tests have been done formally, end users will trust more easily the final product. In addition tests can be done very earlier in parallel with products' development, allowing detecting errors/bugs in the earlier stage of the development.

5 DevOps, TestOps and IoT

IoT implementation in intelligent corporate and residential IT networks poses unique challenges for DevOps as requirements apply well beyond the software development lifecycle and encompasses the complex quality assurance and robust back-end support phase.

Although IoT is largely consumer-driven, the technology is equally pervasive in corporate markets. In this context, DevOps engineers must address traceability and audit ability for all IoT firmware OS developments to ensure compliance success. Collaboration with hardware product specialists and vendors throughout the development process also ensures software robustness to enable streamlined integration with existing IT networks while avoiding vendor lock-in. The world's networking infrastructure with its finite capacity is reaching its limit as the number of IoT endpoints explodes. This in turn, drives interoperability, networking and connectivity issues impacting the wider IT network, whereas IoT development with a focus on network environments, protocols and standards can help eradicate these concerns. Given the scale of IoT production and deployment across the globe, maintaining a robust back-end architecture to automate testing and upgrades requires full visibility into the development cycle as well as a single repository to track changes that follow a device rollout.

Interoperability issues emerge naturally when billions of 'dumb' devices interact with each other. Developing for IoT with the API evolution in mind to expose unique functionalities of the hardware ensures easy rollout of upgrades in addressing integration, connectivity and interoperability issues that may arise down the line.

The performance and behavioral attributes of IoT hardware pose unique challenges for DevOps engineers who must test IoT software in complex real-world environments

and use cases. For instance, weather conditions and durability of the hardware can impact software performance especially when the technology is designed for responding to environmental conditions, such as Web-connected automated fire alarms.

Quality assurance is inherently complex and specialized with the burden of architecture almost entirely falling on the back-end. With this service model, DevOps engineers can push updates frequently as the slow approval process of app stores doesn't hold for IoT software. The IoT ecosystem's vastness has also led to the unpredictability of application requirements for these devices. The understanding of IoT applications is therefore altered even after the launch, prompting significant updates regularly to incorporate the required changes. With the DevOps approach, these updates are directly pushed from the back-end with a continuous delivery service model.

6 Summary and Conclusions

The Internet of Things offers great potential for organizations and societies. A connected world where billions, or even trillions, of devices are connected to the Internet and as such can communicate with each other. If we manage to successfully develop the Internet of Things it will unlock a lot of value and the benefits of the Internet of Things are enormous for organizations and societies. However, there are still some major challenges for the Internet of Things.

Organizations will be able to track their assets in real time, improve utilization of the assets to meet demand. They will be able to predict required maintenance without visiting remote. Monetization of expensive assets becomes easier for organization as the Internet of Things will enable operating expenditure instead of capital expenditure; meaning that users of certain assets are billed based on their actual usage, engine hours, and fuel load etc. instead of having to purchase expensive assets. In addition, devices that are connected to the Internet can receive software updates regularly, instead of replacing the asset thereby prolonging the life of the asset.

The overall challenges in interoperability is first to stabilize the foundation of the real world data services, ensuring technical interoperability from technologies to deliver mass of information and then complementary challenges are for the information to be understood and processed.

The complexity and the diversification embedded in the IoT processes raises many challenges to the Testing organizations in many aspects such as: planning, monitoring, controlling and execution.

The huge amount of things, processes, the big data and complex processes requires a compressive centric testing strategy which will oversees the "Big Picture".

Interoperability testing is a key motive in IoT testing since it addresses the endless amounts of sub systems and its related interactions.

A crucial step for successful integration in a digital world is to reduce test cycle time through the adoption of swift practices and a dynamic test engineering platform. This means fast, responsive QA and testing solutions integrated with agile development.

Choosing the right testing partner can mean the difference between success and failure. Identifying the best practices ensures that products and applications are ready by deadlines and meet customer expectations; ensuring companies deliver defect-free products and services for a quantifiable return on investment.

Companies need to place a strong emphasis on specific cloud and virtualization solutions to create a solid test environment and to manage their cloud and virtualization strategies.

The DevOps approach should also address the disconnect between IT realities and management desires leading to interoperability and productivity concerns for enterprise customers.

The enormous amount of details demonstrated via the endless number of things, processes, sw, hw and SLA may lead to a comprehensive testing strategy which oversees and controls a unified testing life cycle.

Testing is a change agent in the IoT, it provides the natural link between Development and operation from the technological and cultural aspects.

References

1. Vermisan, O., Friess, P.: Building the Hyperconnected Society, IoT Research and Innovation Value Chains, EcoSystems and Markets. RiverSide Publisher, Aalborg (2015)
2. Serrano, M., Barnaghi, P., Carrez, F., Cousin, P., Vermesan, O., Friess, P.: Internet of Things IoT Semantic Interoperability: Research Challenges Best Practices, Recommendation and Next Steps. European Research Cluster on Internet of Things (2015)
3. Security Call in Action, Preparing to the Internet of Things Accenture (2015)
4. D4.1 – Framework for studying existing IoT testing solutions (2013)
5. Internet of Things a Developer's Mandate (2014)
6. TESDT Maturity Model Integration, TMMI Foundation (2012)
7. Certified Tester Foundation Level Extension Syllabus Agile Tester ISTQB (2014)
8. Tester Foundation Level Extension Syllabus Agile Tester ISTQB (2011)