

Taxonomy 4IoT Systems Testing

The Internet of Things (IoT) can be defined as the network of physical objects—“things”—that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. The number of connected devices continues to grow and according to global IoT market forecast, more than 25 billion IoT devices will be connected by 2030. The increase of connected IoT devices, implies the increase of IoT systems deployed. As opposed to traditional software systems, testing IoT systems face many challenges mainly because of heterogeneity, distributivity and dynamic nature of IoT systems, different protocols used, different technologies used, and the needs to test the lower layers such as device or network layer of IoT systems. The existing approaches and tools for testing traditional software testing, may not be enough for testing IoT systems. Many of the professionals involved in testing IoT systems may not have a framework to guide them test better those systems.

We are currently developing testing taxonomy for IoT systems testing to guide the professionals to create a better test strategy with aim of improving test coverage. We also want to make it easier for the practitioners to know different testing types, levels, techniques, and artefacts. Our aim is to ensure that no important aspect of IoT system testing is overlooked while conducting the testing of IoT systems.

To achieve our objective, we reviewed 83 published articles on IoT systems testing and we mined different aspects of IoT systems testing discussed in those articles. We also learned many testing concepts from traditional software testing taxonomy mainly on testing types. In this document, we included all the categories and their possible sub-categories of different aspects of IoT systems testing. We aim to get the feedback of practitioners in terms of its completeness, its usability, and understandability by completing our survey here:

https://docs.google.com/forms/d/e/1FAIpQLSfYstoUH1vINvrH5TpFU2LhO9crGbnibsYbAjuUAEfjL2GgVQ/viiewform?usp=sf_link

We also aim to share the final copy of the revised taxonomy with all the practitioners who wants to keep this as guiding document for testing different IoT systems.

1. Main Categories of IoT Testing Taxonomy

As shown in Fig. 1, the main categories included in this taxonomy are:

1. Testing Approach
2. Testing Metric
3. Testing Target
4. Testing Tools
5. Testing Artifacts
6. Testing Environment

Each of the above category, has sub-categories. We show those sub-categories in the separate figure for better readability of the document.

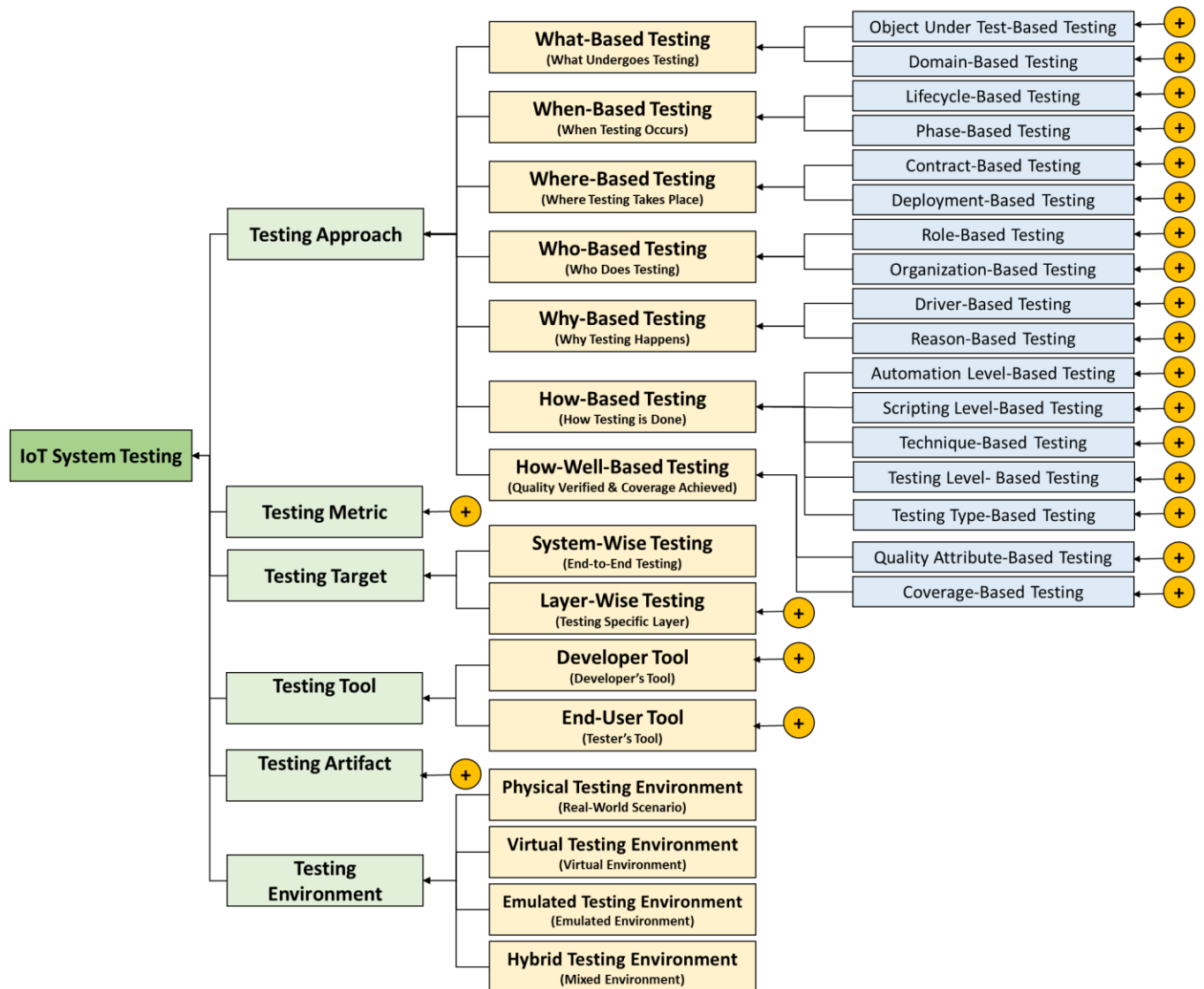


Figure 1: Main Categories in IoT Testing Taxonomy

1.1. Testing Approach

Testing approach for IoT systems refers to the methods and procedures used to ensure that Internet of Things (IoT) systems are functioning correctly and meeting end-users' expectations. This is the main aspect discussed in this taxonomy and it has 7 sub-categories in form of 5Ws and 2Hs.

1.1.1. What-Based Testing

Our focus is to know exactly what to be tested because IoT system consists of many components. We have identified different components that can be tested as shown in Fig 2.

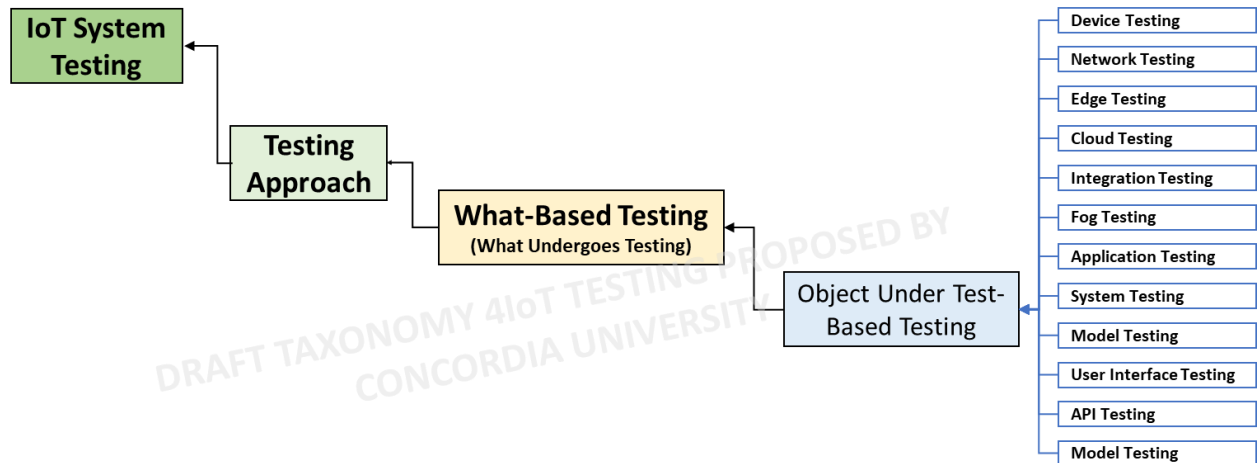


Figure 2: Possible Components to Undergo Testing

Other testers can focus on specific domain. In the Fig. 3, we summarized the possible domains of IoT applications because some tools, approaches, practices, can be domain dependent.

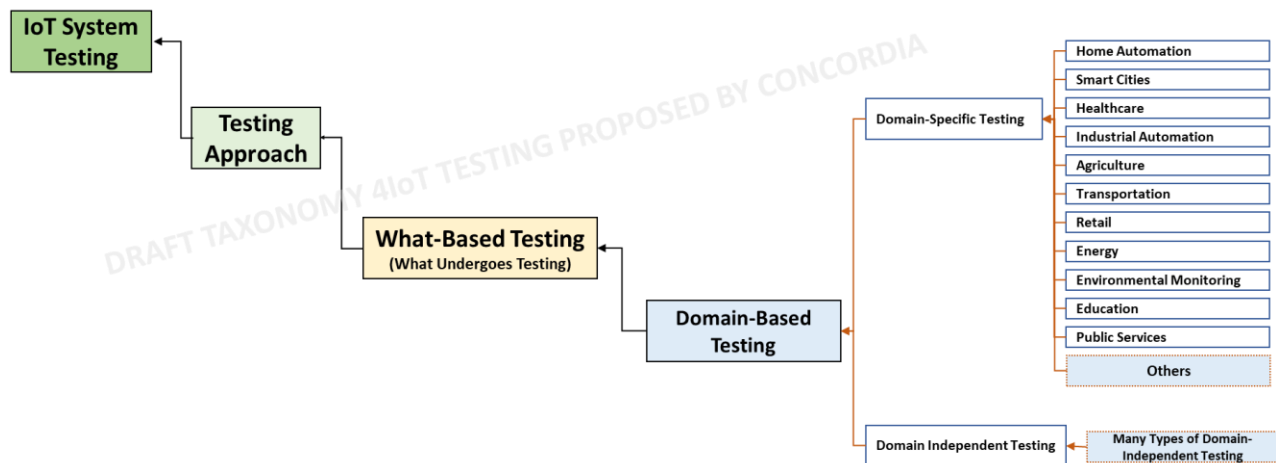


Figure 3: IoT Application Domains to Undergo Testing

1.1.2. When-Based Testing

In this category, we focus on when testing takes place either based on development lifecycle or on specific phase of the project. We have identified 3 sub-categories for lifecycle-based testing as shown in Fig. 4.

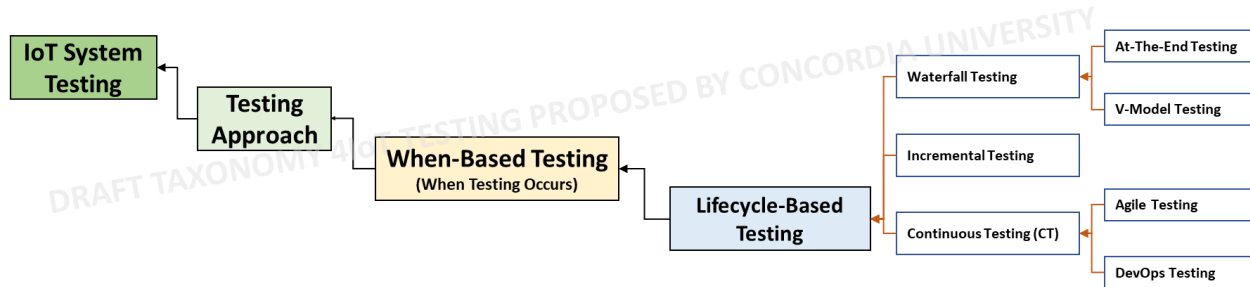


Figure 4: Lifecycle-Based Testing

On phase-based testing, we have identified model-testing which can be done on the requirements and system design phase. Developmental testing when the team is developing, acceptance testing before the system is handed over to the client. Operational testing when the system is in operation.

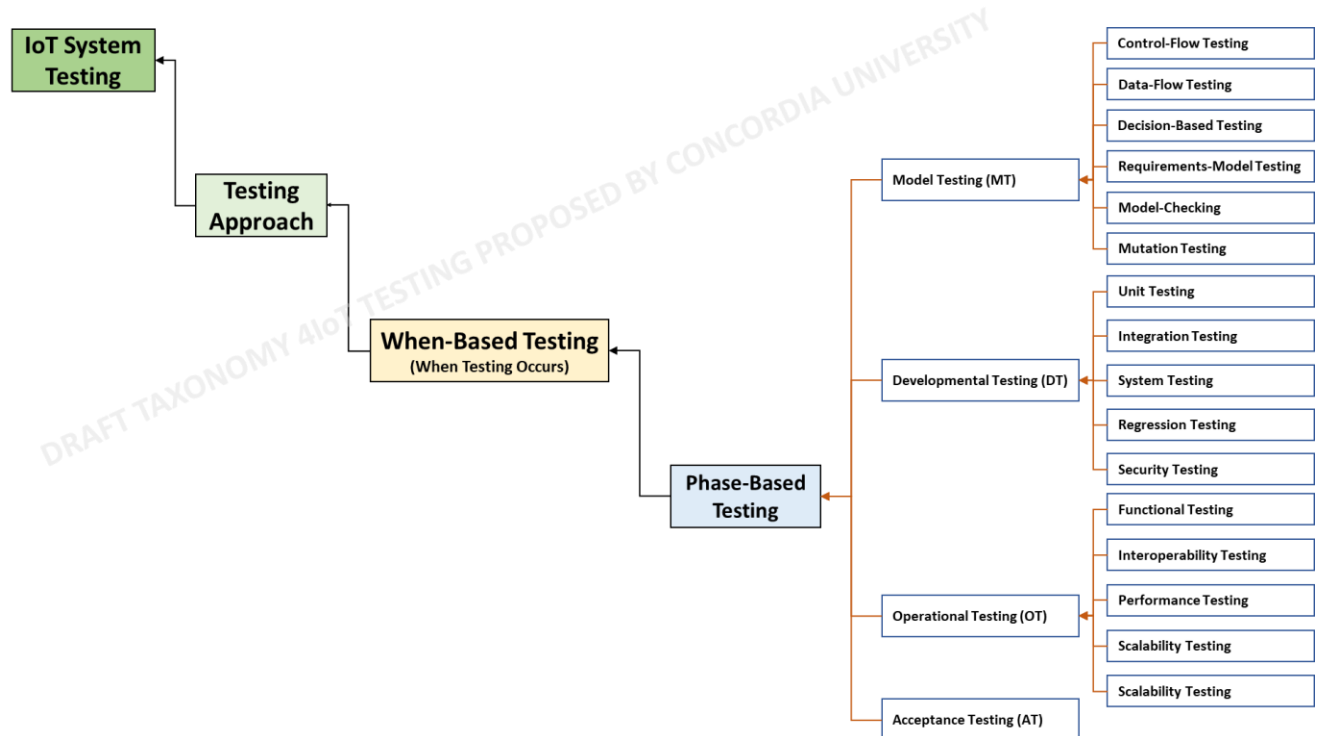


Figure 5: Phase-Based Testing

1.1.3. Where-Based Testing

Under this category, we focused on where the testing is carried out. We identified two sub-categories: contract-based testing where the testing can be done by inhouse team, and outsourced testing when testing is conducted by outside people. Deployment-wise, we identified that testing can be done using cloud resources. Testing can also be performed on multiple systems simultaneously across multiple locations. Testing can also be performed on local device or network as shown in Fig. 6.

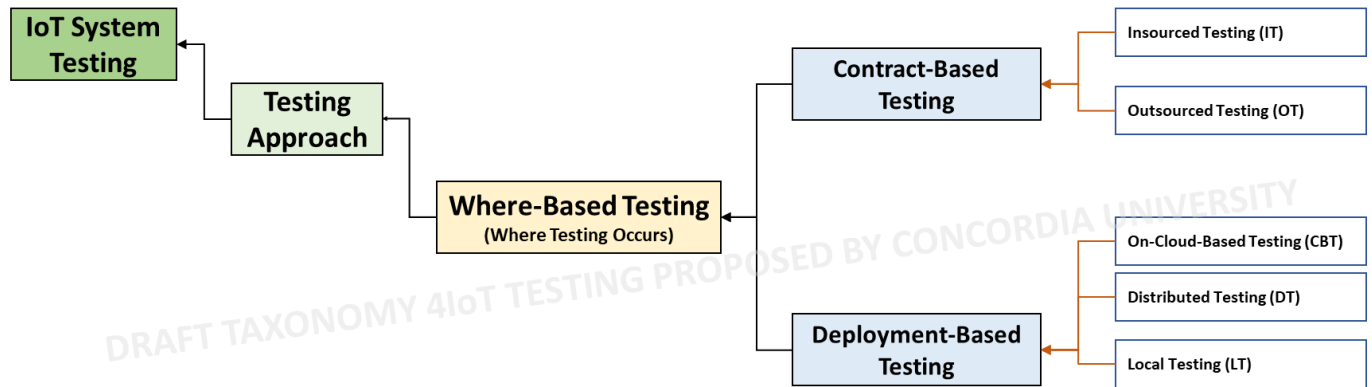


Figure 6: Where Testing is conducted

1.1.4. Who-Based Testing

This category focuses on who does testing. We have identified two sub-categories: Role-based and organization-based testing. Different roles are listed as shown in Fig. 7.

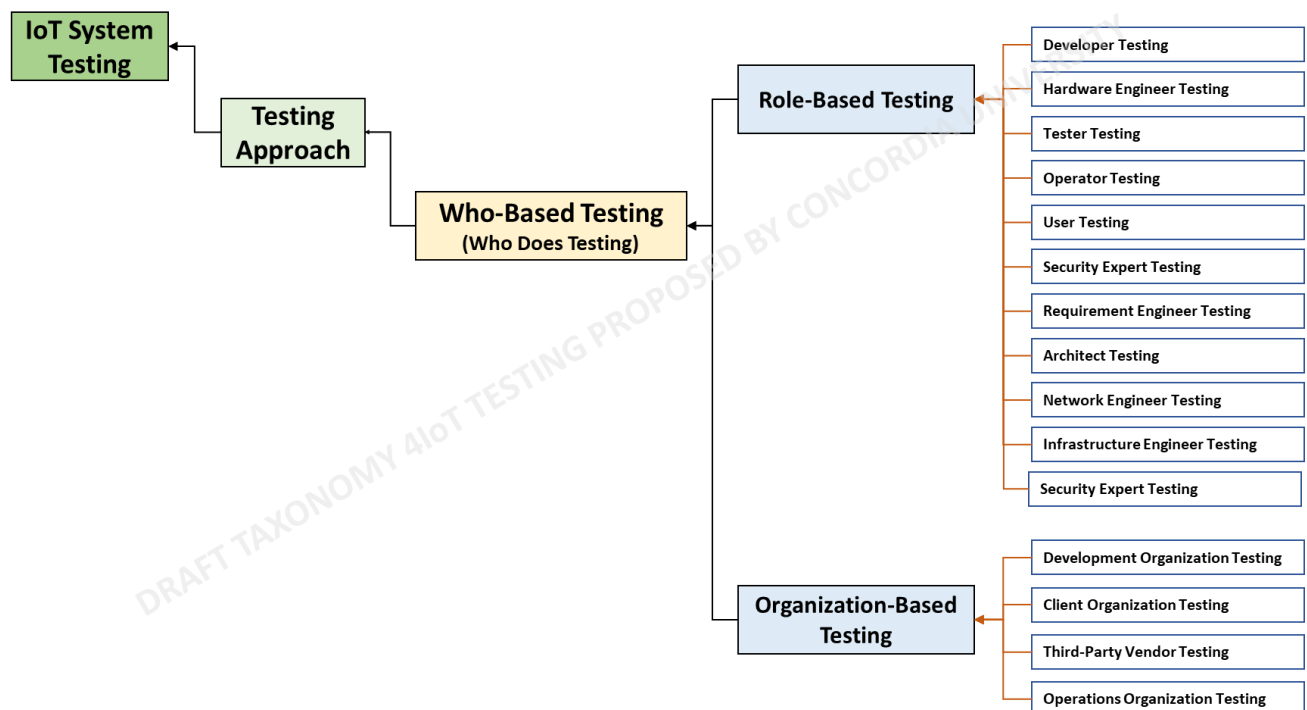


Figure 7: Who does the testing.

On organization-based, the developing organization can do the test. The client if different from developing organization can do the test. Testing can be done by third-party vendor or operations management organization.

1.1.5. Why-Based Testing

This category focuses on why testing happens. We identified that testing can happen for a reason such as initial testing or first time testing, retesting or regression testing. Testing also can happen because of some drivers as summarized in Fig. 8.

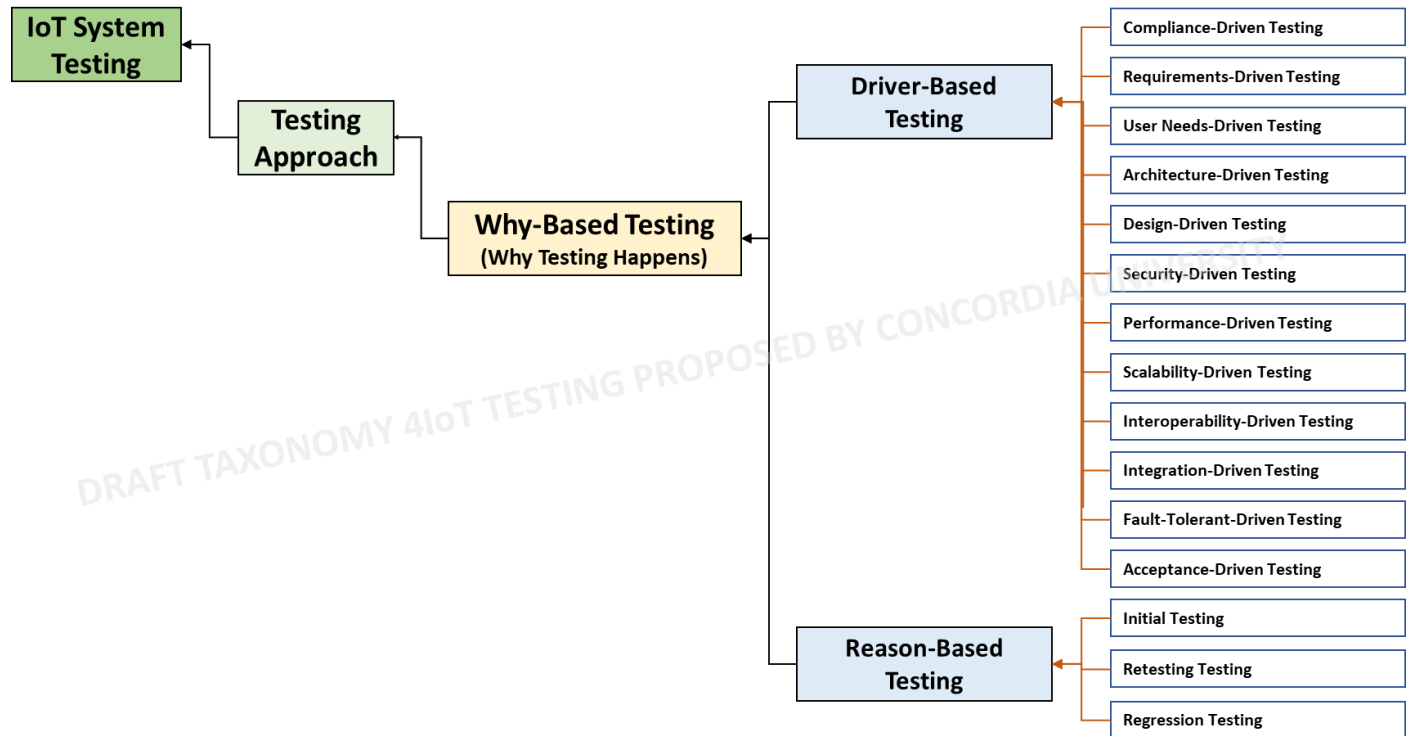


Figure 8: Why Testing Happens

1.1.6. How-Based Testing

This is the main category where different techniques, testing types, and levels are summarized together with all possible sub-categories as per the Fig.9. The detailed definition of each category or sub-category will be provided in the final version of the taxonomy.

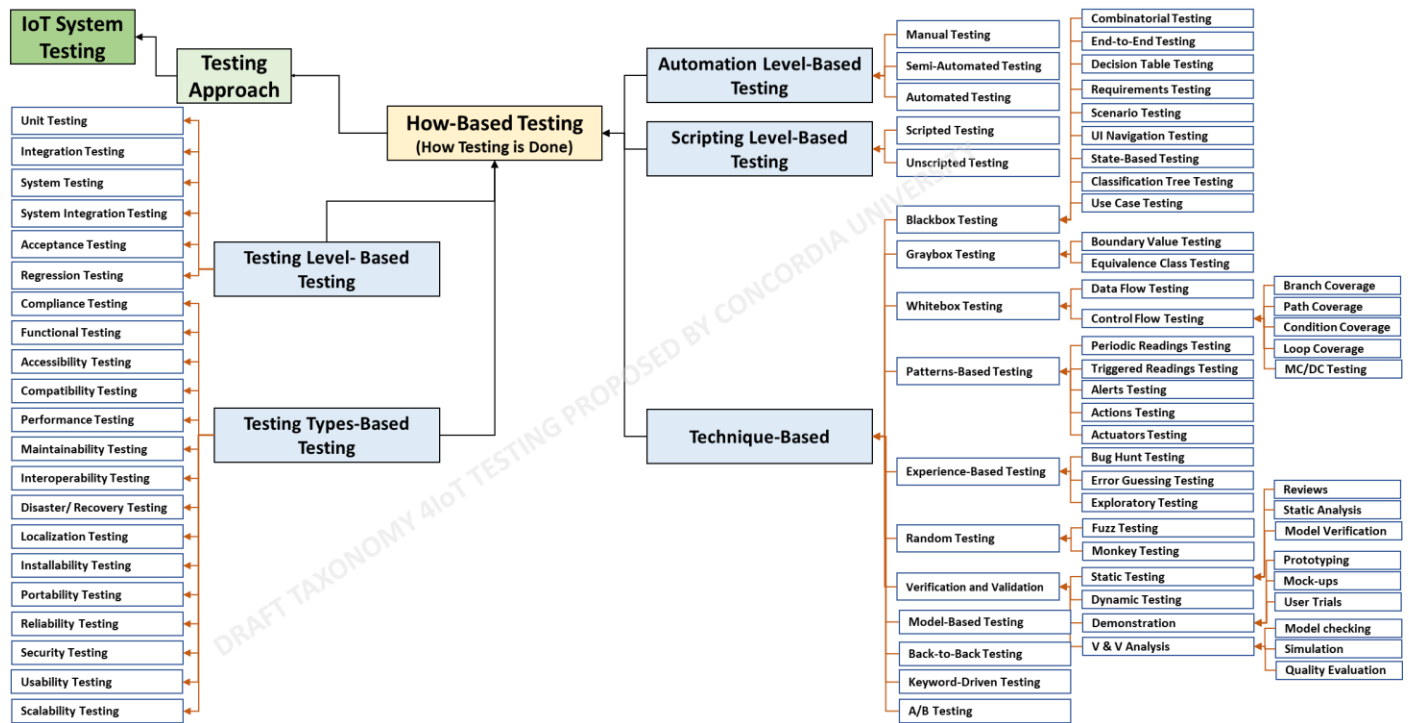


Figure 9: How Testing Is Done

1.1.7. How-Well-Based Testing

To assess how well IoT system is tested, we focused on quality attributes and coverage. We summarized our findings in Fig.10.

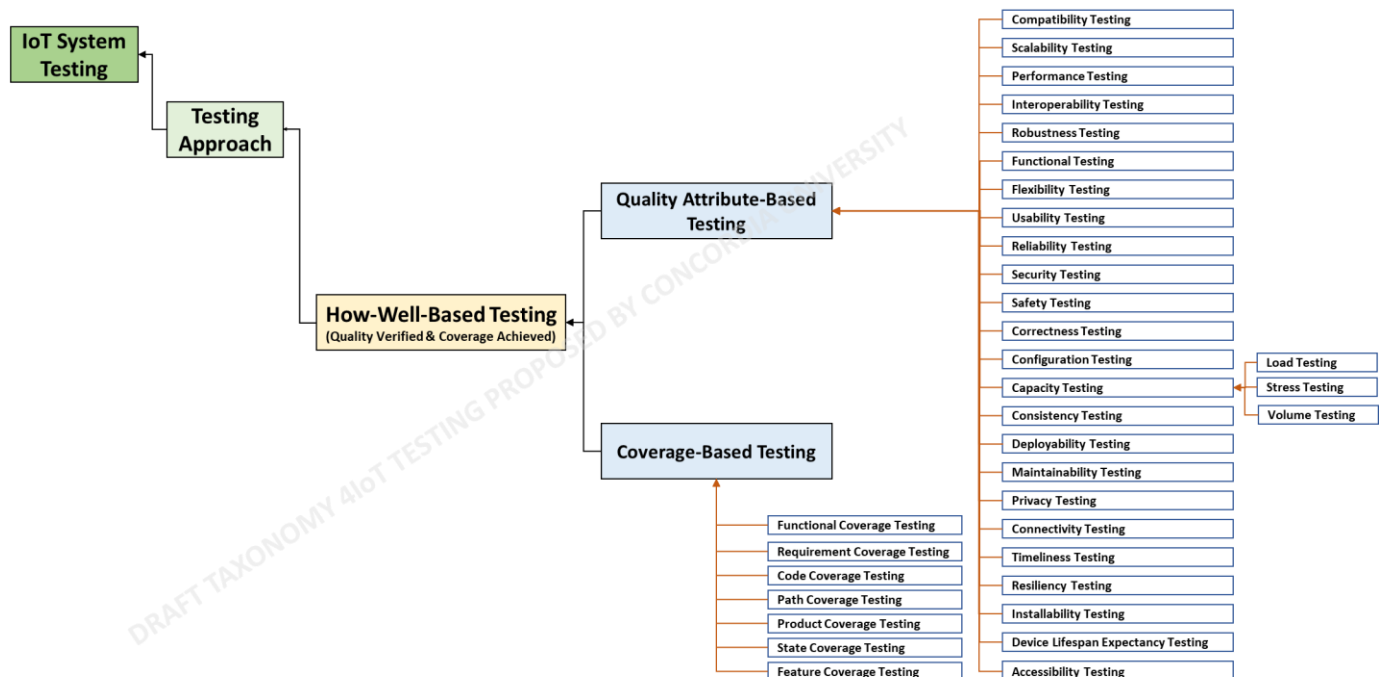


Figure 10: How well IoT System is Tested

1.2. Testing Metric

Fig. 11 summarizes the testing metrics for IoT system testing. We classified those metrics in two major categories: generic and domain-specific. In generic category, we can group them based on code based or functionality based. We categorized the metrics such as performance, connectivity, security, etc., in another sub-category named “Others”. We did not provide any details domain specific metrics because it is not possible to those metrics. Therefore, we can leave this to the testing team for specific domain to identify those metrics.

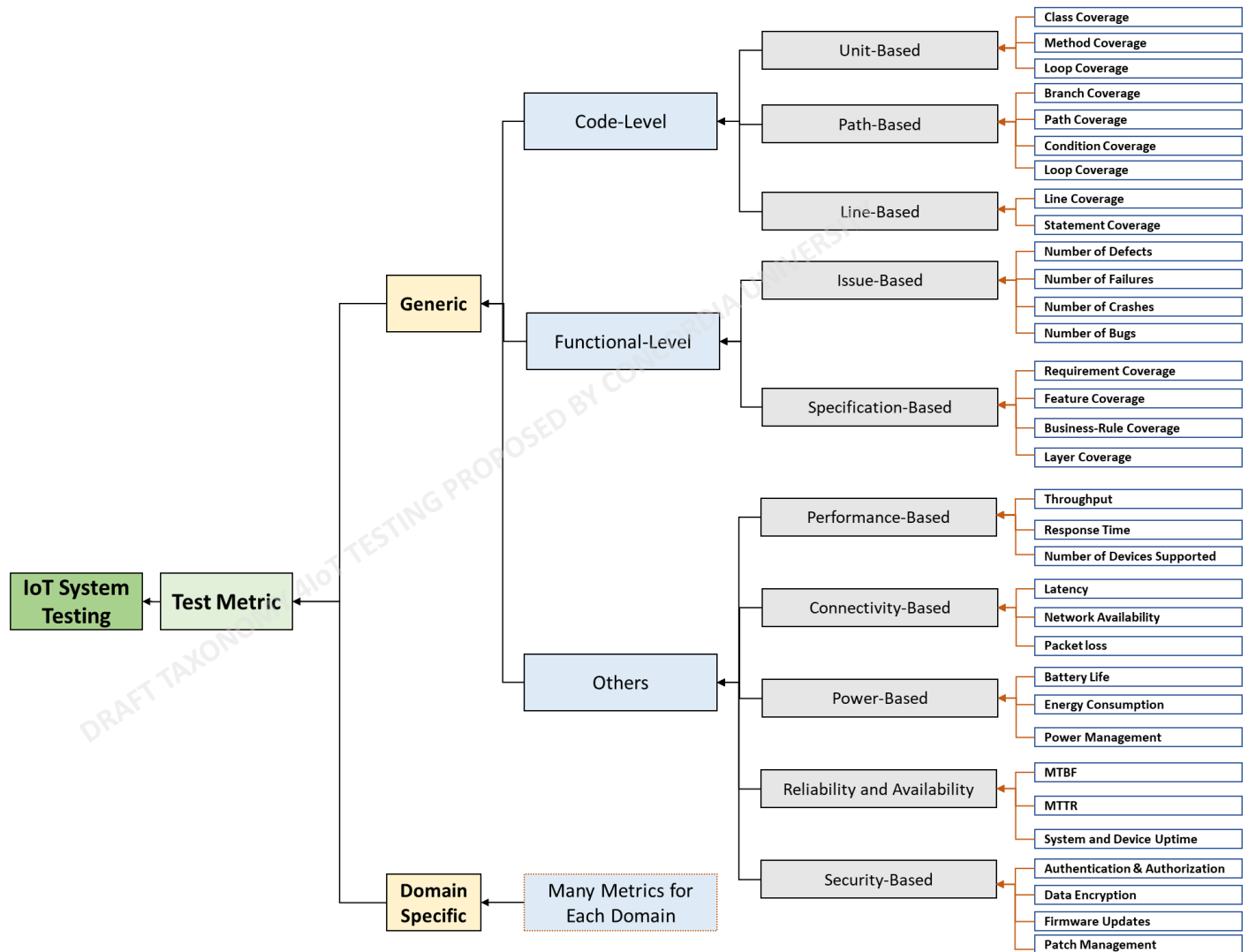


Figure 11: Testing Metrics for IoT Systems

1.3. Testing Target and Tools

1.3.1. Testing Target

This sub-section highlights the components to be considered while testing. Testers can test the entire system or can test specific layer(s) of IoT system as indicated in Fig. 1. When specific testers decide to test specific layer, we focused on main layers that we thought any IoT system will have.

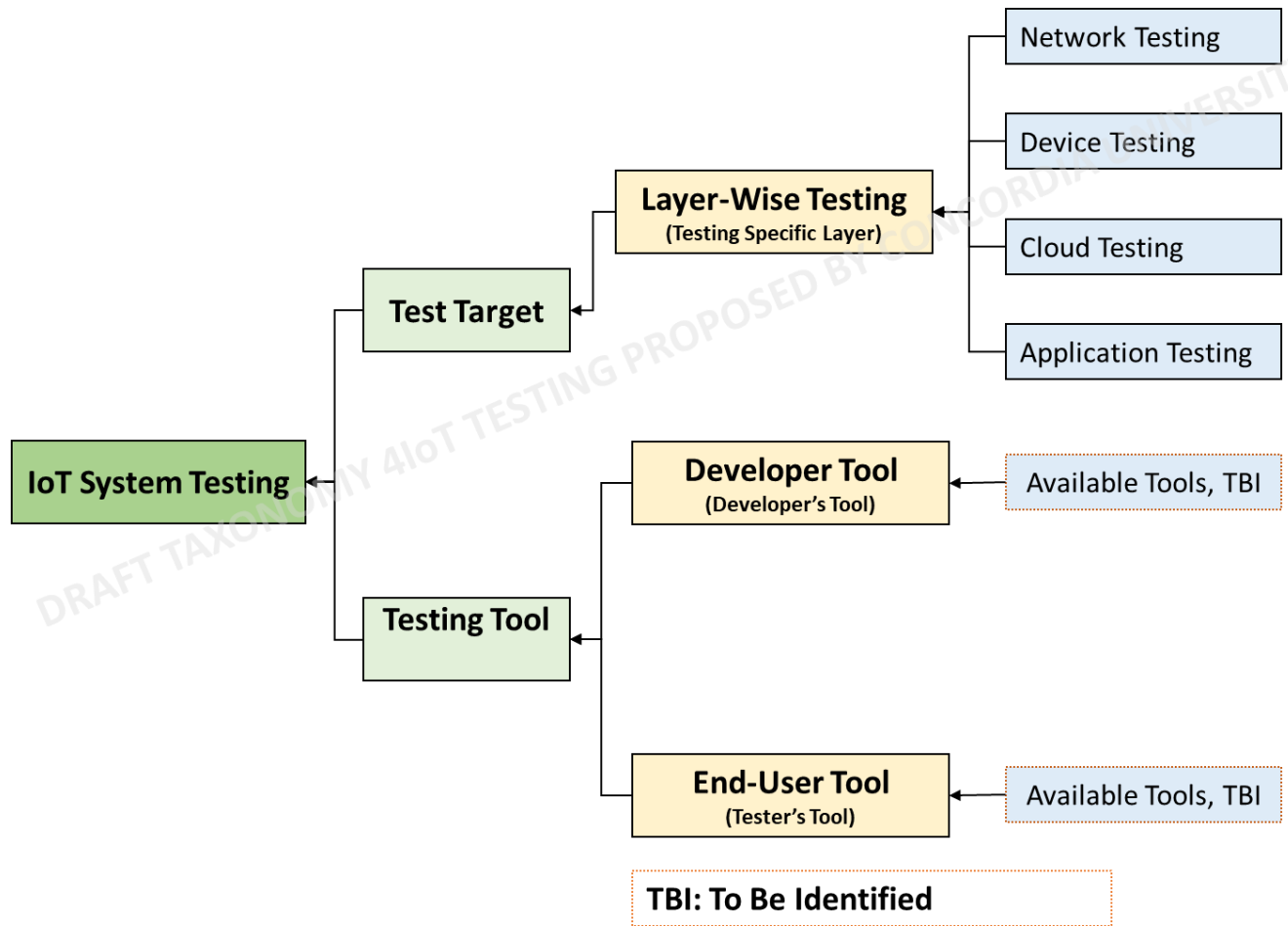


Figure 12: Testing Tools and Test Target

1.3.2. Testing Tools

We categorized testing tools as developer's tool and end-user's tool as indicated in Fig. 12. We did not get exhaustive list of the tools for either category and also, we leave this open for the development and testing team to decide on what tools are available for them. It is worth mentioning that most of the tools available for testing traditional software testing such as selenium, Appium, Apache JMeter, Robot Framework, etc., have been used in testing IoT systems. Node-RED also has been used in many studies for testing IoT system.

1.4. Testing Artifacts

We identified the possible testing artifacts as shown in the Fig. 13. Test Cases, Test Data, Test Script, Defect Report, and Test Report being the commonly mentioned artifacts in the literature. We are not aware of any other artifacts that practitioners consider for IoT systems testing.

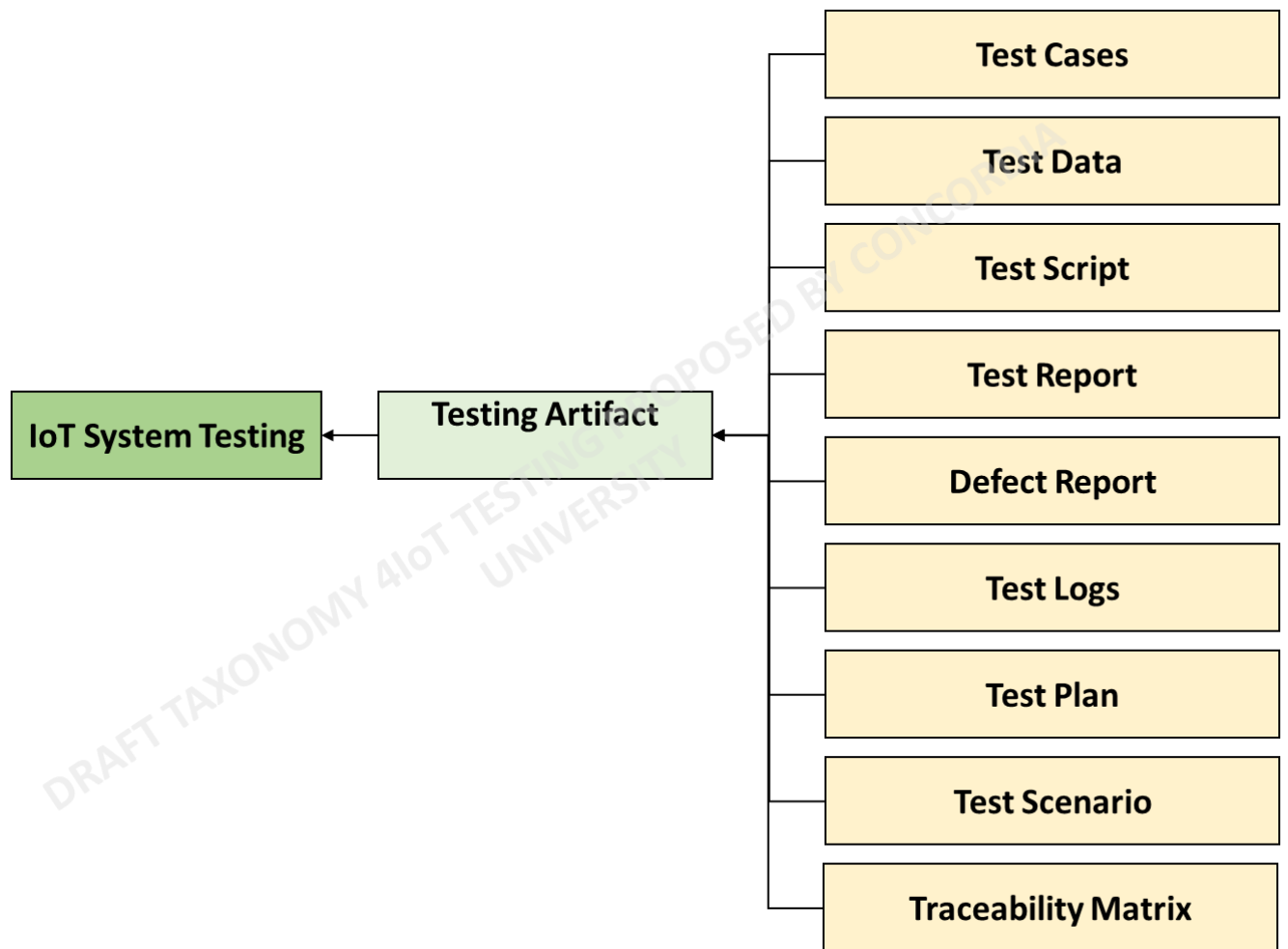


Figure 13: Testing Artifacts

1.5. Testing Environment

All possible testing environments for IoT systems are included in the Fig. 1.