# Syntactic and Semantic Similarities and Discrepancies between Terms of Glossaries for Software Testing

**Guido Tebes[1], Philip Lew[2], Luis Olsina[1]**

[1]GIDIS_Web, Facultad de Ingeniería, UNLPam, General Pico, LP, Argentina

[2]XBOSoft, Woodbridge, VA, USA

{guido_tebes,olsinal}@ing.unlpam.edu.ar, philiplew@gmail.com

***Abstract.*** *For software testing, this work performs a comparison and analysis of syntactic and semantic similarities and discrepancies between 3 glossaries. To conduct the study, 8 terminological categories were conceived, which were used to categorize each glossary term, considering the intended semantics. Also, to count the occurrence frequency of a term in the glossaries, a tool was built that also takes into account the matching of synonyms. Then, the analysis of similarities and discrepancies, as well as absent terms for a subset of them, is performed using metrics and expert interpretations. This study identifies several disagreements in standard terminologies that should merit further attention and efforts to promote harmonization amongst the authors/ publishers of these glossaries with the overarching end goal of assisting their readers in learning and understanding the domain of software testing.*

## 1. Introduction

Today, the software has penetrated every aspect of life. Marc Andreessen's famous quote "Why software is eating the world"[1] has become more than true. Also, many other similar quotes were declared; Mobile is eating the world, Apps are eating the world, Data is eating the world, etc., but the common theme is that we have become more dependent on software. Even traditional hardware-based companies like stereo speakers and garage door openers have been forced to become software companies as software integrates into their hardware devices and platforms.

But what about the quality of the software that has become so important to us? As new development methodologies and business models drive faster software delivery, what happens to the quality? Normally, when you drive a car faster, you have a higher probability of an accident. Hence, even though the software industry has become more mature in the last five decades, the artifacts produced still require more attention with regard to quality. The huge number of produced artifacts can range, for example, from software applications and technology infrastructures to supporting and training documentation. Software glossaries fall within the latter category of artifacts.

A glossary includes agreed entries, i.e., terms and their definitions –and occasionally synonyms, acronyms, and relevant notes- considering significant sources in a given domain. Glossaries certainly serve as a reference to establish a common ground

[1] https://www.ciodive.com/news/software-industry-marc-andreessen/605301/

for terms and definitions not only in learning and understanding but in communicating with others.

Thus, an entire profession and field have been focused on improving software engineering processes and any professional in the field of software engineering will have come across and used many glossaries, either in their formal training or in their daily work. In particular, most professionals in the field of software testing are familiar to some degree with glossaries such as ISTQB (International Software Testing Qualifications Board, (ISTQB, 2021)), TMMi (TMMi, 2018) and ISO 29119-1 (ISO/IEC/IEEE 29119-1, 2013).

Certifications in the software testing field and formal training are often based on some of these glossaries, and in light of this, it is important that these glossaries be of high quality. But what is quality when it comes to evaluating a glossary?

As a starting point, and without giving a complete answer to the previous question, the authors of this work carried out a syntactic and semantic analysis of the similarities and discrepancies of the terms in the three cited glossaries. Through a systematic categorization of terms along with an analysis of them, we have identified inconsistencies in these glossaries that we hope will benefit the profession in eliminating confusion and misunderstanding amongst their readers/users while also providing the authors of these glossaries a foundation for improvement.

Ultimately, we believe our categorization, measurements, metrics, and analysis techniques, can be utilized not only for the current three glossaries that we have analyzed but also for examining the quality of other glossaries from other professions in general.

The rest of the paper is organized as follows. Section 2 outlines some features and numbers of the three standard glossaries analyzed. Section 3 discusses the rationale for dividing testing domain terminologies into conceptual blocks or categories. Section 4 shows the qualitative and quantitative comparison and analysis of syntactic and semantic similarities and discrepancies between the three standard glossaries, highlighting the inconsistent use of the word “testing”. Section 5 provides a summary of related work and discussion. Finally, Section 6 contains the conclusions and future work.

## 2. Overview of the Three Included Software Testing Glossaries

We chose these three glossaries because they are all focused on software testing. Even though their usage context, intended purpose and audience varies, they provide a common foundation and intersection of terms all related to software testing. This gives us common characteristics that can be extracted and analyzed as part of our exploratory study.

The ISO 29119-1 glossary (ISO/IEC/IEEE 29119-1, 2013) is part of a series of standards to be used by an organization when performing software testing as a reference for the other parts of the standard. All terms are thus included to assist those reading and interpreting the five parts of the standard by introducing concepts and vocabulary as a basis for understanding. TMMi (Test Maturity Model integration) is a reference model to support organizations to “improve their software and system testing and achieve higher and sustainable levels of product quality for the systems they are developing and maintaining. With TMMi, these organizations can assess and improve their test processes and, if required, become formally certified" (TMMi, 2018). Hence, the TMMi glossary is intended to support organizations in their test process improvement efforts. ISTQB (International Software Testing Qualifications Board) is a training and certification organization. Thus, its glossary (ISTQB, 2021) is intended to help those taking training and certification syllabus in understanding software testing and specifically to obtain certifications.

As mentioned, each of the glossaries has a different purpose and context, hence different size, scope, and audience related to the other materials in context. For example, ISTQB is intended to assist individuals whereas both ISO 29119-1 and TMMi are intended to assist larger organizational entities. While individuals studying for certifications may have different needs than organizations looking for definitions as a means for collaborative discussion, many of the terms intersect and have different usages as well as synonyms.

There have been other works that analyze glossaries, their structure, and characteristics, but from what we found, other research focuses on one glossary (Arnicane et al., 2016) rather than a comparison of like or similar glossaries, as we discuss later on.

As can be seen in Table 1, we did an initial analysis simply to understand the size and scale of each of the glossaries. The first metric, Total Number of Unique Terms per Glossary in the second row depicts that the ISTQB glossary is overwhelmingly larger than the other two. You might expect this since this glossary is intended to be all-encompassing in the field of software testing for individuals desiring professional certifications. The other two glossaries, intended to be used by organizations in the industry either in the act of software testing or attempting to improve their test processes, are lighter by several orders of magnitude. The third row shows the number of synonyms in each of the glossaries, and the last row shows the total of unique terms plus synonyms to denote the total scale.

We will examine these metrics and many others in subsequent sections.

**Table 1. Basic numbers for the three software testing glossaries**

| Metric name/acronym | ISO 29119-1 | TMMi | ISTQB |
|---|---|---|---|
| Glossary Standard version/year | 1st Ed./2013 | v1.2/2018 | v3.5/2021 |
| Total Number of Unique Terms per Glossary (#UTxG) | 88 | 279 | 588 |
| Number of Synonyms per Glossary (#Sy= #TwithSxG - #UTxG) | 17 | 4 | 160 |
| Total Number of Terms with Synonyms per Glossary (#TwithSxG) | 105 | 283 | 748 |

## 3. Terminological Categories for Terms in Software Testing

Despite the many attempts to standardize testing terms structured in glossaries by different official and *de facto* initiatives, such as ISO, TMMi, and ISTQB, as well as attempts to document testing terms, properties, and relationships structured in ontologies by different researchers as recently analyzed in Tebes et al., (2020), there is often a lack of a broad consensus in the software testing literature and among practitioners on the explicit definition of the terms and their usage. Regarding software testing glossaries, Arnicane et al., (2016) found quality issues in the ISTQB glossary related to consistency, completeness, and correctness. Instead of this study focusing only on a glossary, the present work attempts to obtain evidence for syntactic and semantic similarities and discrepancies between the three aforementioned glossaries for a subset of categorized terms.

To perform the study, eight terminological categories were conceived, which were used to categorize each glossary term, considering the semantics intended by the authors of the standards. The inclusion of terms in categories was carried out at first independently by the authors of the present work, and then we met many times to verify consistency. As a result of this verification via video streaming, some issues were raised and categorization discrepancies in the placement of terms according to the given semantics were agreed upon and resolved.

Table 2 shows the eight terminological categories we designed for domain terms in software testing glossaries. The terms included in categories 1 (C1) to 6 (C6) are specific to the testing domain. Additionally, C7 encompasses terms somewhat related to testing, while C8 includes terms beyond the testing domain that pertains to broader fields such as software engineering or quality.

**Table 2. Names of the eight terminological categories for terms in software testing glossaries**

| Category ID | Terminological Category name |
|---|---|
| C1 | Test Project-, Strategy-, Organizational Test-related Terms |
| C2 | Testing Work Process-, Activity-related Terms |
| C3 | Test Goal-, Requirements-, Entity-related Terms |
| C4 | Test Work Product-related Terms (e.g. Artifact, Report, Result, Specification) |
| C5 | Testing Method-, Technique-, Procedure-, Rule-related Terms |
| C6 | Testing Agent-, Role-, Tool-related Terms |
| C7 | Other Terms somewhat related to Test (e.g., Anomaly, Defect, etc.) |
| C8 | Terms beyond the Test Domain related to Quality or Software Engineering |

The main rationale for designing categories C1 to C6 is as follows. After analyzing both the results of the conducted Systematic Literature Review of primary studies on conceptualized software testing ontologies (Tebes et al., 2020) and the state-of-the-art testing-related standards, we decided to develop a software testing top-domain ontology named TestTDO (Tebes et al., 2021). In the process of defining the ontology scope using competency questions, we found it helpful to devise conceptual blocks for them. From these blocks, we now design the categories C1 to C6 shown in Table 2. The key terms used in the label of each category name are terms or properties in TestTDO, so the reader can refer to their definitions in Tebes et al., (2021) or, for a quick reference, in http://arxiv.org/abs/2104.09232.

Let's briefly analyze C2 and C5. Category 2 is labeled “Testing Work Process-, Activity-related Terms”. It is intended to include glossary terms with the semantics of the testing process, activity or task. In other words, the terms testing process, activity, or task encompass the meaning of ‘what to do’ rather than ‘how to do’ a testing activity. Instead, Category 5 is devoted to including testing method terms, which have the semantics of ‘how to do’ a testing activity/task description. C5 is labeled “Testing Method-, Technique-, Procedure-, Rule-related Terms” in Table 2. For example, the term Testing Method is defined in Tebes et al., (2021) as “a specific and particular way to perform the specified steps for a task included in a Testing Activity”. The explicit semantic distinction between glossary terms that represent ‘what to do’ and ‘how to do’ has a clear benefit for understanding. For instance, for the same Design Testing activity (what to do), different Testing Design Methods (how to do) can be assigned (Tebes et al., 2021). About a decade ago, Henderson-Sellers et al., (2014) noted that a highly visible and challenging definition that is urgently needed to ensure that ISO standards are consistent with other industry usage is that of ‘method' versus ‘process’.

When searching terminological categories for software testing glossary terms, we found two works in the literature. One of them is the test classification of the ISTQB glossary represented in a recent draft document. It has the following categories: Testing, Software Engineering, Requirements, Quality, and General. Note that the Testing category is the only one specific to the testing domain; although the Requirements category includes some terms specific to testing as well. Thus, in order to categorize glossary terms, we have designed six categories as stated above, in which terms for Test Requirements fall in C3. Therefore, for the rest of the ISTQB categories, we have conceived C7 and C8. It is worth noting that the scope of a glossary is generally a bit broader than the domain of software testing. According to the authors of ISTQB: “Some related non-testing terms are also included if they play a major role in testing, such as terms used in software quality assurance and software lifecycle models”.

The other somewhat related work for categorization carried out by Kuļešovs et al., (2013) aims at structuring testing ideas into eight classes. For example, the class called “How to test (approach, method, technique)?” corresponds mainly to C5 and, to a lesser extent to C1, in which the testing approach and strategy-related terms are placed. However, an explicit class for C2 (Testing Work Process-, Activity-related Terms) is missing in Kuļešovs et al., (2013).

As a result of our classification, the reader can find the ISO glossary terms categorized in Appendix II of the document at http://bit.ly/ComplementaryResults. The TMMi glossary terms and the ISTQB glossary terms are classified in Appendixes III and IV, respectively.

Table 3 shows the numbers of classified terms and percentages for categories 2 and 5 of the three analyzed glossaries. The numbers for all categories and proportions are in Appendix V of the aforementioned linked document.

**Table 3. Numbers and percentages for Categories 2 and 5 of the three software testing glossaries. Recall that the values for #UTxG (Total Number of Unique Terms per Glossary) are in Table 1**

| Metric name/acronym | ISO 29119-1 | TMMi | ISTQB |
|---|---|---|---|
| Number of Unique Terms per Glossary for Category2 (#UTxGC2) | 45 | 43 | 103 |
| Percentage of Unique Terms per Glossary for Category2 [%UTxGC2 = (#UTxGC2 / #UTxG) * 100] | 51.14% | 15.41% | 17.52% |
| Number of Unique Terms per Glossary for Category5 (#UTxGC5) | 2 | 27 | 46 |
| Percentage of Unique Terms per Glossary for Category5 [%UTxGC5 = (#UTxGC5 / #UTxG) * 100] | 2.27% | 9.68% | 7.82% |

The next section deals with the qualitative and quantitative comparison and analysis of syntactic and semantic similarities and discrepancies between the analyzed standard glossaries only for C2 and C5.

The underlying hypothesis in this work is that considering both the syntactic and semantic aspects of the terms according to the authors of the glossaries, many of those that fell into C2 should be in C5, and vice versa. In other words, there are syntactic and/or semantic inconsistencies between terms of glossaries found in C2 (what to do) and C5 (how to do), which could hinder understandability.

To carry out this exploratory study, we calculated the Total Sum of Unique Terms considering the summed values shown in Table 1. This result is in the first row of Table 4. Then, the sum of unique terms for the categories C2 and C5 is 266, which represents 27.85% of the total of unique terms (955). It is important to remark that C8, which includes 334 terms beyond the testing domain represents 34.97%.

**Table 4. Summed values of the three software testing glossaries just for Categories 2 and 5. Recall that the #UTxG (Total Number of Unique Terms per Glossary) values are in tables 1 and 3**

| Metric name/acronym | Value |
|---|---|
| Total Sum of Unique Terms [TUT = (#UTxISO + #UTxTMMi + #UTxISTQB)] | 955 |
| Total Sum of Unique Terms for Category2 [TUTC2 = (#UTC2xISO + #UTC2xTMMi + #UTC2xISTQB)] | 191 |
| Total Sum of Unique Terms for Category5 [TUTC5 = (#UTC5xISO + #UTC5xTMMi + #UTC5xISTQB)] | 75 |
| Total Sum of Unique Terms for Categories2&5 [TUTC2&5 = (TUTC2+ TUTC5)] | 266 |
| Percentage of Unique Terms for Categories2&5 [%TUTC2&5 = (TUTC2&5 / TUT) * 100] | 27.85% |

## 4. Analysis of Glossary Terms considering Syntactic and Semantic Similarities and Discrepancies

This Section illustrates our analysis results of syntactic and semantic similarities and discrepancies amongst the three glossaries as shown in subsection 4.2. It is important to remark that we have developed a tool to support the syntactic frequency calculation between the three glossaries. The followed procedure by the tool to calculate syntactic matching between glossary terms is documented in subsection 4.1. Finally, subsection 4.3 documents other issues detected in the glossaries such as absences and inconsistencies, among others.

### 4.1. Procedure to Syntactically Match Terms Between Glossaries

When we started to collect all terms and their corresponding synonyms in each software testing glossary we observed several situations. For example, the ISO glossary has the Organizational Test Strategy term, the TMMi glossary has the Test Strategy term and the ISTQB glossary also has the Test Strategy as a main term in addition to the term Organizational Test Strategy as a synonym of Test Strategy. In analyzing this situation, we have concluded that the term Test Strategy (or Organizational Test Strategy as a synonym) has a syntactic frequency of 3. This result means that the term Test Strategy is present in the three analyzed glossaries. We have concluded this result since we consider that if we have a term with one or more synonyms, we take into account the name of the term or any of its synonyms to calculate the syntactic comparison with another term (or synonym) from another glossary. In other words, in the previous example, the ISTQB glossary terms named Test Strategy or Organizational Test Strategy are used interchangeably in the frequency calculation but count as a single term.

Another situation we had was the following. The ISTQB glossary has the main term "white-box testing" with the following synonyms: clear-box testing, code-based testing, glass-box testing, logic-coverage testing, logic-driven testing, **structural testing**, and structure-based testing. In addition, the ISO glossary has the main term "structure-based testing" with the following synonyms: **structural testing**, glass-box testing, and white box testing. Finally, the TMMi glossary has only the term "white-box testing" without synonyms.

In conclusion, the term "white-box testing" has a syntactic frequency of 3. One way to obtain this result is that "white-box testing" has as a synonym the term "structural testing" in the ISTQB glossary, and the ISO glossary has the term "structural testing" as a synonym of the "structure-based testing" term. Therefore, we have a syntactic matching between the terms of these two glossaries. In addition, the TMMi glossary has the term "white-box testing" which syntactically matches with the term "white-box testing" of the ISTQB glossary and, therefore, the TMMi glossary "white-box testing" term syntactically matches with the term structure-based testing of the ISO glossary or any of its synonyms (structural testing, glass-box testing, and white box testing) by transitivity between glossary terms and synonyms. Note that we can obtain the same result (i.e., a frequency of 3) in different ways, for example considering the "structure-based testing" synonym in ISTQB and the main term "structure-based testing" in ISO. Additionally, at this point, it is important to remark that we consider removing the hyphens in the terms for the syntactic analysis. Note that ISO has the term "white box testing" and the other glossaries have the term "white-box testing".

Therefore, considering the above examples, we have used the following rule when we calculated the syntactic frequency between glossary terms: *Let's suppose that we have the term T1 in the glossary G1, and T2 is a synonym of T1. Also, we have the term T2 in the glossary G2, and T3 is a synonym of T2. Then, if we have the term T3 in the glossary G3, the term T1 syntactically matches with the term T3 by transitivity of terms and synonyms between glossaries. Therefore, T1 (or T2 or T3) has a syntactical frequency of 3.*

We have developed a tool that follows this rule to automatically calculate the syntactic matching between glossary terms and their synonyms.

### 4.2. Analysis of Syntactic and Semantic Similarities and Discrepancies

Once obtained the results of the syntactic frequency for each glossary term by using the procedure described in subsection 4.1, we calculated the numbers and percentages shown in Table 5. Note that we used a set of metrics, as described below.

The first metric shown in Table 5 is the Number of Terms with Frequency 3 in Categories2&5 (#TFq3C2&5). Note that frequency 3 implies a syntactic similarity of the same term, considering the synonyms, in the three glossaries, e.g., we found the same term "static testing" in the three glossaries. Also, we found 15 terms more with a frequency of 3, so the #TFq3C2&5 is 16 in total.

**Table 5. Metrics and their values for the terms' syntactic frequencies of the three software testing glossaries. Recall that TUTC2&5 (Total Sum of Unique Terms for Categories2&5) = 266 according to Table 4**

| Metric name/acronym | Value |
|---|---|
| Number of Terms with Frequency 3 in Categories2&5 (#TFq3C2&5) | 16 |
| Percentage of Terms with Full Syntactic Similarity in Categories2&5 [%TFSySC2&5 = ((#TFq3C2&5*3) / TUTC2&5) * 100] | 18.05% |
| Number of Terms with Frequency 2 in Categories2&5 (#TFq2C2&5) | 48 |
| Percentage of Terms with Partial Syntactic Similarity in Categories2&5 [%TPSySC2&5 = ((#TFq2C2&5*2) / TUTC2&5) * 100] | 36.09% |
| Number of Terms with Frequency 1 in Categories2&5 (#TFq1C2&5) | 122 |
| Percentage of Terms without Syntactic Similarity in Categories2&5 [%TwSySC2&5 = (#TFq1C2&5 / TUTC2&5 ) * 100] | 45.86% |
| Percentage of Terms with Full Syntactic Similarity for All categories C1-C8 (%TFSySforAll) | 13.51% |
| Percentage of Terms with Partial Syntactic Similarity for All categories C1-C8 (%TPSySforAll) | 32.67% |
| Percentage of Terms without Syntactic Similarity for All categories C1-C8 (%TwSySforAll) | 53.82% |

Then, we calculated the Percentage of Terms with Full Syntactic Similarity in Categories2&5 (%TFSySC2&5) and it resulted in 18.05%. At this point, it is important to remark that the metric %TFSySC2&5 uses the value obtained in #TFq3C2&5 multiplied by 3 since we have 3 terms for each term with frequency 3. Also, the total amount of terms in the calculated percentage is 266 corresponding with TUTC2&5 since we consider only C2 and C5 in this work.

Additionally, we have calculated the same metrics but considering all categories (C1-C8) and we obtained that the Percentage of Terms with Full Syntactic Similarity for All categories C1-C8 (%TFSySforAll) is 13.51% ≅ ((43*3)/955)*100 (note that 955 corresponds with TUT in Table 4).

Analogously to what we did for terms with syntactic frequency 3, we did the same for terms with syntactic frequency 2 and 1. For frequency 2, the Number of Terms with Frequency 2 in Categories2&5 (#TFq2C2&5) is 48. This implies that a term in a certain glossary syntactically matches with another term of only one of the other 2 remaining glossaries. For example, the term “acceptance testing” is in ISTQB and TMMi glossaries, and the term “accessibility testing” is in ISTQB and ISO glossaries. Hence, the Percentage of Terms with Partial Syntactic Similarity in C2 and C5 (%TPSySC2&5) is 36.09% (((48*2)/266)*100). Additionally, the Percentage of Terms with Partial Syntactic Similarity for All categories C1-C8 (%TPSySforAll) is 32.67% (((156*2)/955)*100).

The reader can see the other obtained values for frequency 1 in Table 5. Note that a term with frequency 1 implies that the term name is only in one of the glossaries.

If we compare the obtained results shown in Table 5, we can conclude that the three glossaries have few terms included in C2 and C5 with full syntactic similarity (18.05%) and in general (for all categories) as well (13.51%). Moreover, most glossaries terms in C2 and C5 have a frequency of 1 (45.86%) and, considering all categories, also most terms have a frequency of 1 (53.82% ≅ (514/955)*100).

**Table 6. Metrics and their values for syntactic frequencies and semantic similarities/discrepancies of glossaries' terms are taken from the data processed and recorded in Appendix VI at http://bit.ly/ComplementaryResults. Recall that TUTC2&5 (Total Sum of Unique Terms for Categories2&5) = 266 according to Table 4**

| Metric name/acronym | Value |
|---|---|
| Number of Terms with Full Semantic Similarity for Frequency 3 in Categories2&5 (#TFSSFq3C2&5) | 11 |
| Number of Terms with Partial Semantic Similarity for Frequency 3 in Categories2&5 (#TPSSFq3C2&5) | 5 |
| Number of Terms with Full Semantic Similarity for Frequency 2 in Categories2&5 (#TFSSFq2C2&5) | 47 |
| Number of Terms without Semantic Similarity for Frequency 2 in Categories2&5 (#TwSSFq2C2&5) | 1 |
| Percentage of Total Terms with Full Syntactic and Semantic Similarity in Categories2&5 [%TTFSSSC2&5 = (#TFSSFq3C2&5 * 3/ TUTC2&5) * 100] | 12.41% |
| Percentage of Total Terms with Partial Semantic Similarity in Categories2&5 [%TTPSSC2&5 = ((#TPSSFq3C2&5 * 2 + #TFSSFq2C2&5 * 2)/ TUTC2&5) * 100] | 39.10% |
| Percentage of Total Terms without any Semantic Similarity for Categories2&5 [%TTwSSC2&5 = ((#TPSSFq3C2&5 + #TwSSFq2C2&5 * 2 + #TFq1C2&5) / TUTC2&5) * 100] | 48.49% |

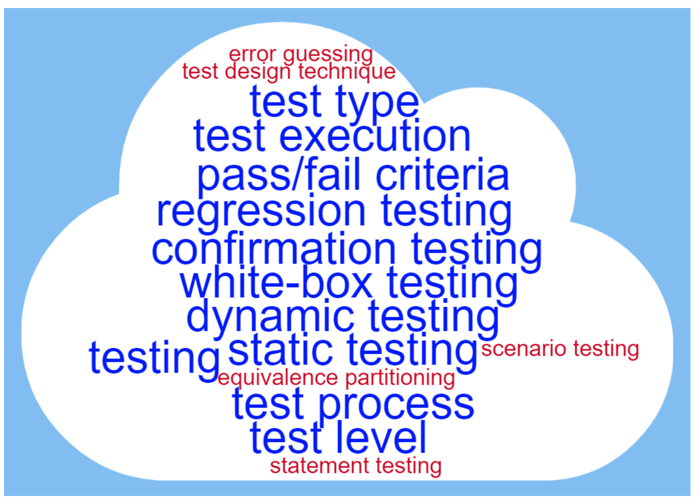


**Figure 1. Word cloud for glossary terms with a full syntactic similarity that includes 16 terms with a syntactic frequency of 3 in Categories 2 and 5. Among them, the largest terms (11 blue terms) have full semantic similarity, while the smallest terms (5 red terms) have partial semantic similarity**

On the other hand, in Table 6, we illustrate the results of metrics related to semantic similarities. Regarding the semantic similarities and discrepancies of the 16 terms with a syntactic frequency of 3, only 11 terms have full semantic similarity, i.e., the 3 syntactically same terms in the three glossaries have the same intended semantics as well. The remainder 5 terms have a partial semantic similarity, i.e., they have a semantic similarity of only 2 terms out of 3.

We show in Figure 1 a word cloud that illustrates the names of the abovementioned 16 terms with a syntactic frequency of 3. Note that the biggest size terms (blue highlighted) have a full semantic similarity while the smaller ones (red highlighted) have a partial semantic similarity. For example, the term “white-box testing” has a full semantic/syntactic similarity, and the term “test design technique” has a full syntactic similarity but partial semantic similarity.

In order to explain what a “semantic matching” between two terms from 2 glossaries with the same syntax means in this work, we will use Table 7.

As shown in Table 7, the term “white-box testing” (or structure-based testing) is a kind of “dynamic testing” in ISO. Also, the term “dynamic testing” is a kind of “testing”, which in turn is a set of activities. Then, we conclude that testing, dynamic testing, and structure-based testing fall in C2 since they are terms related to processes and activities according to their definitions. Something similar happens for the term “white-box testing” in the other 2 glossaries.

Therefore, the term “white-box testing” falls in the same category for the three glossaries. In addition, if we analyze the 3 definitions in-depth, we can conclude that the three glossaries mention the structure of the test object (i.e., the test item, system or component) and therefore the intended semantics of the term is the same for the three glossaries according to our judgment.

**Table 7. Definitions of the “white-box testing” and “test design technique” terms in the three analyzed glossaries, as well as some related terms with their definitions**

| Term | Definition | Glossary | Category |
|---|---|---|---|
| structure-based testing | **Dynamic testing** in which the tests are derived from an examination of the **structure** of the **test item**. *Recall that “structure-based testing” is a synonym of “white-box testing” in ISO 29119-1.* | ISO 29119-1 | C2 |
| white-box testing | **Testing** based on an analysis of the **internal structure** of the **component or system**. | ISTQB | C2 |
| white-box testing | **Testing** based on an analysis of the **internal structure** of the **component or system**. | TMMi | C2 |
| dynamic testing | **Testing** that requires the execution of the test item. | ISO 29119-1 | C2 |
| testing | **Set of activities** conducted to facilitate discovery and/or evaluation of properties of one or more test items | ISO 29119-1 | C2 |
| testing | **The process** consisting of all lifecycle activities, both static and dynamic… | ISTQB | C2 |
| testing | **The process** consisting of all lifecycle activities, both static and dynamic… | TMMi | C2 |
| test design technique | **activities**, concepts, **processes**, and patterns used to construct a test model that… | ISO 29119-1 | C2 |
| test technique | A **procedure** used to define test conditions, design test cases, and specify test data. *Recall that “test technique” is a synonym of “test design technique” in ISTQB.* | ISTQB | C5 |
| test design technique | **Procedure** used to derive and/or select test cases. | TMMi | C5 |

On the other hand, if we do the same analysis for the term "test design technique", we conclude that the intended semantics for the term in the TMMi and ISTQB glossaries is technique/procedure and, therefore, falls in C5, unlike ISO glossary that has process/activity semantics (C2).

In the literature is recognized that "white-box testing" and "black-box testing" are 2 common terms broadly used in the software testing community. We were surprised by the fact that the term "black-box testing" does not have a syntactic frequency of 3. We investigated this situation and we realize that the term "black-box testing" with the semantics of process/activity (C2) appears only in the ISO and TMMi glossaries, but in the ISTQB it only has the term "black-box test technique" with the semantics of method/technique (C5).

### 4.3. Other Detected Issues

When we look at the results of the Percentage of Terms without Syntactic Similarity in Categories2&5 (%TwSySC2&5 = 45.86%) and Percentage of Terms without Syntactic Similarity for All categories C1-C8 (%TwSyS-forAll = 53.82%) in Table 5, we can note that most glossaries' terms do not have a syntactic similarity with other glossary terms.

This issue implies, on the one hand, a large absence of terms in the glossaries. A cause of this is the different proportions of glossaries' terms. As shown in Figure 2 (a), ISTQB contributes to TUTC2&5 (Total Sum of Unique Terms for Categories2&5 in Table 4) with more than half of all the glossaries' terms. Likewise, as shown in Figure 2 (b), the same happens for the glossaries' terms considering all categories (C1-C8).

On the other hand, another cause of this problem is that we found some terms with the same semantic but with different syntax, and therefore these terms have a syntactic frequency of 1. Just to mention a few examples, the terms "test completion" in ISTQB and "test completion process" in ISO; "maintenance testing" (ISTQB) and "maintainability testing" (ISO); "test monitoring" (ISTQB) and "test monitoring and control process" (ISO); among others.

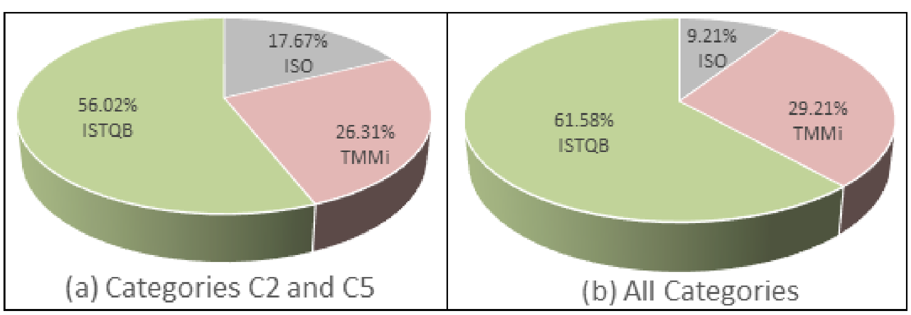


**Figure 2. The ratio of glossaries' terms for C2 and C5 (a); and for all categories C1-C8 (b)**

Another detected issue is related to the inconsistent use of the term "testing". The three glossaries consider that the term "testing" has the semantics of process/activity (i.e., related to C2). Table 8 shows in the %TTeTC2xG (Percentage of Total Terms that end with the word "Testing" in Category2 per Glossary) metric that most of the terms ending with the "testing" word in their names (or synonymous) fall in

C2 in these glossaries. Recall that we classified the terms by analyzing their given semantics and not the term name.

However, only ISO uses the "testing" word consistently (%TTeTC2xG = 100%). ISO includes 28 terms that end with the word "testing" in the term name and, considering that the "testing" term definition has the semantics of activity/process, these 28 terms fall in the category C2 accordingly. The same does not happen in the other two glossaries, since the metric %TTeTC2xG gives 62.07% in TMMi and 75% in ISTQB. This could lead to interpretation problems and ambiguities in these glossaries.

Although ISO uses the "testing" word consistently in C2, we noted in the definition of the term "test design technique" that it has the given process/activity semantics when it should have the method/technique semantics (i.e., related to C5 as TMMi and ISTQB did). Besides, ISO has the terms "statement testing" and "scenario testing", and we categorized them in C2 since their definitions mention that are a kind of "test design technique" and therefore, considering that in ISO semantically a "test design technique" falls in C2, then these 2 terms fall in C2 as well.

However, if the given semantics of "test design technique" were more coherent, some terms such as "statement testing" and "scenario testing" would fall in C5 and ISO will not be 100% consistent with using the word "testing" in the terms' names.

**Table 8. Metrics and their values related to terms' names that end with the word "testing" in Categories 2 and 5**

| Metric name/acronym | Values per Glossary | | |
|---|---|---|---|
| | ISO 29119-1 | TMMi | ISTQB |
| Total Number of Terms that end with the word "Testing" in Categories2&5 per Glossary (#TeTC2&5xG = #TeTC2xG + #TeTC5xG) | 28 | 29 | 84 |
| Total Number of Terms that end with the word "Testing" in Category2 per Glossary (#TeTC2xG) | 28 | 18 | 63 |
| Percentage of Total Terms that end with the word "Testing" in Category2 per Glossary [%TTeTC2xG = (#TeTC2xG / #TeTC2&5xG) * 100] | 100% | 62.07% | 75% |
| Total Number of Terms that end with the word "Testing" in Category5 per Glossary (#TeTC5xG) | 0 | 11 | 21 |
| Percentage of Total Terms that end with the word "Testing" in Category5 per Glossary [%TTeTC5xG = (#TeTC5xG / #TeTC2&5xG) * 100] | 0% | 37.93% | 25% |

## 5. Related Work and Discussion

To the best of our knowledge, no directly related work in the literature considers a comparative analysis of syntactic and semantic similarities and discrepancies for a set of software testing glossaries.

In order to look at related work in digital libraries, we primarily searched Scopus with a variety of keywords and operators, even including glossaries outside the software testing domain. The result was less than 10 papers, which two authors of this work analyzed in depth. Among them, the most relevant research was carried out by Arnicane et al., (2016). Contrary to our research, they analyzed only inconsistencies in one software testing glossary, without conducting a comparative analysis.

In Arnicane et al., (2016) the authors detected many syntactic and semantic issues in the ISTQB glossary. For example, they detected that the terms “test process” and “testing” have the same semantics in ISTQB, that is, one of them can be supposed as a redundant term in the glossary and they should be synonymous. We noted that the same situation happens in the other 2 glossaries.

Focusing on the results evidenced in Section 4 and taking into account the underlying hypothesis stated in Section 3, we would like to outline at least one simple suggestion that can promote harmonization, homogeneity, and ultimately quality improvement.

Regarding the syntactic aspect of naming terms in C2 (what to do) and C5 (how to do), we recommend a clear distinction between them, for example, adding the word “technique” to some terms in C5. Thus, “regression testing” has the meaning of process/activity in all three glossaries due to the definition of the word “testing” (recall Table 7), and the intended semantics given to it by the authors of the glossaries. So, this is reasonable, but, “statement testing” has the method/technique/procedure semantic in ISTQB and TMMi, so we suggest syntactically disambiguating it by using the term “statement testing technique”. Instead, in ISO there is also a semantic inconsistency, as noted at the end of subsection 4.3.

## 6. Concluding Remarks and Future Work

This paper has presented the results of analyzing the syntactic and semantic similarities and discrepancies between terms of three analyzed glossaries for software testing documented in ISO 29119-1 (ISO/IEC/IEEE 29119-1, 2013), TMMi (TMMi, 2018) and ISTQB (ISTQB, 2021). We have supported this analysis by using a set of metrics in addition to a set of categories which helped us to semantically categorize all glossaries' terms.

It is important to point out that the analysis carried out in this paper focused only on categories 2 and 5, namely: software testing terms for work processes/activities and method/techniques/procedure, respectively. In conclusion, as shown in Table 6, we found few syntactic/semantic matching between the terms of the three analyzed glossaries. Also, among other issues, we noticed an inconsistent use of the word “testing”, as we have documented in subsection 4.3.

Recalling the hypothesis mentioned in Section 3, which states that many terms with the semantics of process/activity (category 2 –what to do) should have the semantics of method/technique (category 5 –how to do it) and vice versa, we can confirm that hypothesis based on the obtained results.

Therefore, we recommend to the authors of the glossaries make a clear distinction between terms with the semantics of process/activity (e.g., those terms that finish with the word “testing”) and terms with the semantics of method/technique/procedure, which should finish with the words “testing technique” or “testing design technique”. We argue that by making this distinction explicit, the glossaries can foster terminological homogeneity and improve understandability when learning concepts.

In future work, we will include category 1 (C1, Test Project-, Strategy-, Organizational Test-related Terms) in the analysis of the glossaries' terms. We already observed that exist some terms that end with the word “testing” in the term name that fall in C1. This issue could cause misunderstanding in the glossary reader since the term “testing” has the semantics of process/activity, and in C1 the reader would expect to find terms with the semantics of strategy/approach/project/organization.

Additionally, in future work, we will put more emphasis on the sub-characteristics and attributes of information quality that can be considered for evaluation and comparison, such as semantic correctness, completeness of coverage, and non-redundancy of coverage, syntactic and semantic consistency, among others.

**Acknowledgments.** This line of research is supported partially by the Engineering School at UNLPam, Argentina, in the project coded 09/F079, and named “Family of Strategies for Functional and Non-Functional Software Testing considering Different Test Goal Purposes”.

## References

Arnicane, V., Arnicans, G., & Borzovs, J. (2016). Building of Concept System to Improve Systematic Collection of Terminology. *Frontiers in Artificial Intelligence and Applications*, volume *291*, 313–326. https://doi.org/10.3233/978-1-61499-714-6-313.

Henderson-Sellers, B., Gonzalez-Perez, C., McBride, T., Low, G. (2014). An ontology for ISO software engineering standards: 1) Creating the infrastructure. *Computer Standards & Interfaces*, *36*(3), 563–576. https://doi.org/10.1016/j.csi.2013.11.001.

ISTQB. (2021). International Software Testing Qualifications Board. *ISTQB®: Standard Glossary of Terms used in Software Testing*, version 3.5. https://www.istqb.org/

ISO/IEC/IEEE 29119-1. (2013). *Software and systems engineering – Software Testing – Part 1: Concepts and definitions*.

Kuļešovs, I., Arnicane, V., Arnicans, G., & Borzovs, J. (2013). Inventory of Testing Ideas and Structuring of Testing Terms. *Baltic Journal of Modern Computing*, *1*(3–4), 210–227.

Tebes, G., Olsina, L., Peppino, D., & Becker, P. (2021). Specifying and Analyzing a Software Testing Ontology at the Top-Domain Ontological Level. *Journal of Computer Science & Technology (JCS&T)*, *21*(2), 126–145. https://doi.org/10.24215/16666038.21.e12

Tebes, G., Peppino, D., Becker, P., Matturro, G., Solari, M., & Olsina, L. (2020). Analyzing and documenting the systematic review results of software testing ontologies. *Information and Software Technology*, volume *123*, 1–23. https://doi.org/10.1016/j.infsof.2020.106298

TMMi. (2018). TMMi Foundation. *Test Maturity Model Integration (TMMi®) - Guidelines for Test Process Improvement*, Release 1.2.