Specifying A/B tests: a systematic literature review

Wouter Kok
Vrije Universiteit Amsterdam, The Netherlands
w.j.kok@student.vu.nl

ABSTRACT

Context. Online controlled experiments, also called A/B tests, is an experimental technique where an audience is divided in a control and treatment group, in which the treatment group is assigned to a different variant. The outcome will dictate whether the variant has better results or not. The specifications of A/B tests include important information about how the A/B test should be run, but the field of A/B testing seems young and limited when it comes to specifying A/B tests.
Goal. The aim of this study is to evaluate and analyze the current approaches towards specifying online controlled experiments to increase understanding of the present status of specifying A/B tests in the field. But also to gather this information for future experimenters, developers and researchers in this field.
Method. The method used in this study is a systematic literature review where a qualitative study is done to summarize the state of the art in the specification of online tests.
Results. The results show three main approaches towards specifying A/B tests: experimentation platforms, domain specific languages, and methodologies. The most commonly found way of specifying A/B tests is within experimentation platforms.
Conclusions. The current field of specifying A/B tests is young and therefore only 9 primary studies have been found. From the results of this literature review we can conclude that the different approaches towards specifying A/B tests are experimentation platforms, domain specific languages, and methodologies. Combining the information from these three approaches when specifying A/B tests could be a handhold for developers and experimenters of future A/B tests, although there is a demand for more future research.

KEYWORDS

Systematic Literature Review, A/B Testing, Online Controlled Experiments, Specification

1 INTRODUCTION

Controlled experiments have been carried out for over hundreds of years. One story dating back till the 1700s about a crewman on a ship trying to heal sick sailors with a citrus, but in the 1920s the theory has been started to form by Sir Ronald A. Fisher. [8] [15]

Controlled experiments, also called randomized experiments, A/B tests, split tests, control/treatment, and parallel flights, are experiments where a group is split into two or more groups; a control group and one or multiple treatment groups [15]. The control group gets no treatment, or in the domain of the web, no variant of a web-page. However, the treatment group gets a variant of the existing web-page. The results, often in the form of metrics, are then analyzed to see what the most favorable method is. With the explosion of the web, controlled experiments were being executed online, resulting in the term *Online controlled experiment (OCE).*

Large corporations have the resources to build their own experimentation platform, such as LinkedIn, Facebook and Microsoft [2, 10, 19]. Thousands of experiments are being run on software, websites and even operating systems, to improve user experience and increase revenue [16]. The reason for this literature review, is that the author and supervisor of the author noticed that the current field of A/B tests is overflowing with papers expanding upon challenges, pitfalls, and definitions of A/B testing, including how to do them [5, 9, 14, 15]. However, there seems little information about specifying A/B tests. Therefore, the goal of this research is to analyze the current field of A/B tests, aiming at the specifications of these tests, to discover how A/B tests can be specified and what the differences and commonalities are between approaches.

The organization of the paper is as follows. First the related work is reported to get a grasp of the current field of A/B testing. Then the study design is explained with the formal goal of the research. After that the results of the data synthesis are described. Next a discussion will be provided, ending with threats to validity and the conclusion.

1.1 Background

In this section, a terminology is given for the reader to better understand a few of the terms involved with online controlled experiments. Therefore, the practical guide by Kohavi et al. is used as this covers a broad range of terms [15].
Overall Evaluation Criterion. To formally express the experiment's objective, experimenters set a quantitative measure called the OEC. When setting the OEC, experimenters should be careful not to formalize this measure with short-term goals in mind, because that can lead to problems in the long run.
Factor. A measure for the amount of factors in an experiment, e.g. a normal A/B test has one factor with the values A and B.
Variants. In an A/B test, the control group and treatment group get assigned a different variant, which is a user-experience that is being tested.
Experimental unit. Metrics (for example *clicks* on a web-page) are being calculated over the experimental unit. An example of an experimental unit can be a visitor of a website.
A/A test. A test where the control and treatment group are assigned to the same variant to analyze whether e.g. the experimentation system works correctly without any unusual results.

2 RELATED WORK

There are a multitude of papers about pitfalls, challenges and recommendations regarding A/B testing. Kohavi et al. discussed unexpected outcomes of OCEs [13]. Pitfalls of long-term OCEs have been analyzed by Dmitriev et al. in [5]. The pitfalls that come with the interpretation of metrics in OCEs have been discussed by Dmitriev et al. in [6]. Gupta et al. go over the challenges, best practices and

pitfalls in evaluating the results of OCEs. [9] Also in the automotive sector the A/B pitfalls have been discussed [7]. Therefore, A/B testing is a challenging topic with a lot of things to consider. This paper could be an addition to these papers, in specific to prevent issues arising when specifying A/B tests.

Bigger companies, like Microsoft and LinkedIn, have contributed to the field of A/B testing by describing their experimentation platforms in which experimenters can run experiments such as A/B tests. [10, 16, 19]. Although the process of specifying A/B tests often is a small section of these papers, these papers add useful information to the topic of specifying A/B tests and the overall field of A/B testing.

Furthermore, Facebook has created a Domain Specific Language for OCEs called PlanOut, which is described by Bakshy et al. in [2]. In this language, a test can be written in a PlanOut script, which is serialized into JSON. This language is build to work on Facebook's systems, therefore more use cases may be necessary to apply the language to other systems.

With the current field of A/B testing and OCEs, the specification of A/B tests still hasn't caught the attention that other sub-fields mentioned before have. Hence this systematic literature review will add on to that.

3 STUDY DESIGN

3.1 Research Goal

From the current literature of A/B testing, we want to identify best practices of how A/B tests can be formally specified. Results of this literature review will be analyzed to gain a greater understanding of this concept. It is important to accurately perform the literature review. Hence this review adheres to the *Guidelines for performing Systematic Literature Reviews in Software Engineering* by Kitchenham and Charters [12]. In addition, the Goal-Question-Metric (GQM) approach is adopted to formalize the research goal of this literature review: [3]

Analyze	the research landscape of A/B testing
for the purpose of	evaluation and comparison
with respect to their	specifications
from the point of view of	A/B testers and experimenters
in the context of	optimizing IT products

3.2 Research Questions

The author and the supervisor of this literature study noticed that in the field of online controlled experiments the specification part is often overlooked in other studies (see Section 1 and 2). Therefore the following research question is established to supper the research goal.

RQ-1: *What are the different approaches for specifying online controlled experiments?*

With the term *specifying*, we denote the process of fabricating an online controlled experiment, including the instructions that are needed prior to executing the A/B test, for example describing variables for variants and the OEC. The developed specification can be considered as the blueprint of an A/B test.

3.3 Initial search

All relevant studies need to be identified, therefore a protocol document is used to capture the process of the search for literature. In addition, a Google spreadsheet is used to capture papers that could potentially be primary papers. For managing the found papers, Zotero[1] is used. Both the protocol document and Google spreadsheet can be found in a Github repository given in Section 3.8.

The Google spreadsheet has the following columns: *year, title, abstract, authors, type of publication, publication, DOI, Exclusion and Inclusion criteria, potential, included, digital library, notes, snowballing source.* It has one tab including the papers from the search process, and another tab for the data collection models.

The following search strategy is applied:

(1) After the research question is specified, keywords will be extracted from the research question (but also from the field).
(2) The resulting keywords will be used in the search strings, given in the protocol document (see Section 3.8).
(3) The search strings are applied to the following digital libraries: Science Direct[2], Google Scholar[3], IEEExplore[4], ACM Digital Library[5], Springer Link[6].
(4) First, Google Scholar is put to use, because this contains papers from a variety of different digital libraries. Afterwards, other digital libraries are searched. See the protocol document for more details on the search process.
(5) The search strategy will not include *Manual Search* to avoid potential bias. For this reason, all search results will be saved to Zotero collections named according to the digital libraries.
(6) The title will be checked against the inclusion / exclusion criteria.
(7) If it does, use Zotero and put it in a "Picked" collection, when in doubt in a "Doubt" collection, else in a "Not Picked" collection.
(8) Add the papers of the "Picked" collection to the data extraction sheet.
(9) Check the abstract of the papers and see if they hold to the inclusion /exclusion criteria. If it does, mark "potential" in the data extraction sheet.
(10) At last, do a full paper scan and check again for the inclusion / exclusion criteria. If they hold, check "included?" in the data extraction sheet.
(11) Go through the "Doubt" collection for potential papers and start at 6 again.
(12) The snowballing protocol will be applied to only the primary studies, which is explained in Section 3.5.

One thing to consider are duplications found in the search process. Whenever a duplication was found, criteria E4 (see next Section) was checked to prevent duplicate papers in the primary studies.

[1] https://www.zotero.org/
[2] https://www.sciencedirect.com/
[3] https://scholar.google.nl/
[4] https://ieeexplore.ieee.org/
[5] https://dl.acm.org/
[6] https://link.springer.com/

Table 1: Data Collection Model

Data item	Value
Study Identifier	Integer
Goal of the study	String
Description of the specification approach	String
Example specification	String
Type of the approach	String
Application domain	String

3.4 Application of selection criteria

In the process of selection, inclusion and exclusion criteria are used to check whether a publication has potential to be a primary study. These criteria are specified below:

- I1 The publication is not in the process of peer reviewing and is published (DOI).
- I2 Studies that focus on online controlled experiments with information about the approach.
- E1 The year of publication should be at least 2008.
- E2 The publication should be written in English.
- E3 Studies that focus on online controlled experiments but not in the context of websites or not mentioning approaches.
- E4 A paper that is already included or an extension of a paper that is already included.

3.5 Snowballing

Snowballing can be used to find related papers by searching for references and citations from other papers [12]. In this review, all primary studies were "snowballed" forward and backward twice. Thus if a primary study is found using snowballing, then snowballing can be used on that paper as well, but no further. The data extraction sheet has 1 column which indicates that a paper is found by using snowballing.

3.6 Data Extraction

In the extraction of data from the papers, we look specifically for approaches and specifications towards A/B testing. With this in mind, a *data collection form* is created. This form will be filled in according to each primary study in the data extraction sheet. At first the current landscape of research was unknown and therefore we couldn't structure the data collection form properly. Thus first all papers were analyzed and the A/B specification approach was described. After discussing with the supervisor, a decision was made to divide the papers into the type of approach as there were reoccurring types. With this information, the data collection model was expanded. The final data collection model can be seen in Table 1.

3.7 Data Synthesis

Using the extracted data, there will be qualitatively looked at commonalities and differences between the different specification approaches, but we can not make clear decisions before actually analyzing the data. The studies will be tabulated based on the specification approach used in the study. The first type is a practical approach where the specification is within an experimentation platform. The second type is an approach of defining a domain specific language to specify the A/B tests. The third type of approach is a more theoretical, methodological description of an approach.

Within these approaches, we will look for *similarities*, *differences* and *opportunities*.

3.8 Study Replicability

Here is a link to all data from this literature review:
https://github.com/wkokgit/Specifiying-A-B-Tests-SLR-data

The data consist of the research protocol, the primary study list, data extraction sheet and raw extracted data.

4 RESULTS

The results of the data synthesis of the literature review are reported in this section. From thoroughly reading the primary studies, we discovered a difference in taken approaches. There are three approaches of specifying A/B tests identified: *Experimentation Platform*, *Domain Specific Language*, and *methodology*. Each approach is expanded upon in a subsection. Moreover, each approach will have three subsections as described in the data synthesis in the study design. The author will refer to the papers by their Study ID, e.g. "*PS 62 shows that ..., however PS 3 shows ... .*", where PS stands for primary study. However, this notation is only in the *Results* section to make it more understandable for the reader. But first, the demographics will be shown to give a broader view of the found papers.

4.1 Demographics

The results of the search process are visualized in Figure 1. For each digital library, the amount of taken results is shown; each number is the amount of papers. The found papers in the grey box, can be obtained from the GitHub repository. The papers from the red box are included in the data extraction overview in which the papers from the orange box are papers with potential. Earlier there was mentioned that the aim of the initial search was to find around 20 papers, but at the end, 9 papers were included as primary studies, 5 of them are found through snowballing. The primary studies are shown in Table 2 with the Study ID that is conjointly used in the data collection forms. The authors of the papers are academic and all papers are published in conferences. The last 5 papers have successive Study ID's, due to the snowballing phase. Furthermore, all papers are rather recent, with the oldest paper being from 2014 and the newest from 2020. Figures 2 and 3 show more information about the *type of approach* and *type of application domain*, respectively. Papers demonstrating an experimentation platform are found the most. Moreover, the paper with Study ID 3, was about an Experimentation Platform, but showed a Domain Specific Language as well. In the bar chart showing the type of domains, the majority of approaches come from the web domain.

4.2 Experimentation Platform

One way that leads to having to specify A/B tests, is when creating an experimentation platform. A total of 5 experimentation platforms

Table 2: List of primary studies

Study ID	Year	Study Title	Authors
3	2015	From infrastructure to culture: A/B testing challenges in large scale social networks [19]	Xu, Ya; Chen, Nanyu; Fernandez, Addrian; Sinno, Omar; Bhasin, Anmol
11	2018	An activity and metric model for online controlled experiments [17]	Mattos, David Issa; Dmitriev, Pavel; Fabijan, Aleksander; Bosch, Jan; Olsson, Helena Holmström
42	2018	The anatomy of a large-scale experimentation platform [10]	Gupta, Somit; Ulanova, Lucy; Bhardwaj, Sumit; Dmitriev, Pavel; Raff, Paul; Fabijan, Aleksander
61	2009	Facilitating Controlled Tests of Website Design Changes: A Systematic Approach [4]	Cámara, Javier; Kobsa, Alfred
62	2014	Designing and deploying online field experiments [2]	Bakshy, Eytan; Eckles, Dean; Bernstein, Michael S.
63	2019	Experimentation in the Operating System: The Windows Experimentation Platform [16]	Li, P. L.; Dmitriev, P.; Hu, H. M.; Chai, X.; Dimov, Z.; Paddock, B.; Li, Y.; Kirshenbaum, A.; Niculescu, I.; Thoresen, T.
64	2019	Who's the Guinea Pig? Investigating Online A/B/n Tests in-the-Wild [11]	Jiang, Shan; Martin, John; Wilson, Christo
65	2019	Continuous A/B Testing in Containers [18]	Révész, Ádám; Pataki, Norbert
66	2020	Continuous Experiment Definition Characteristics [1]	Auer, F.; Lee, C. S.; Felderer, M.

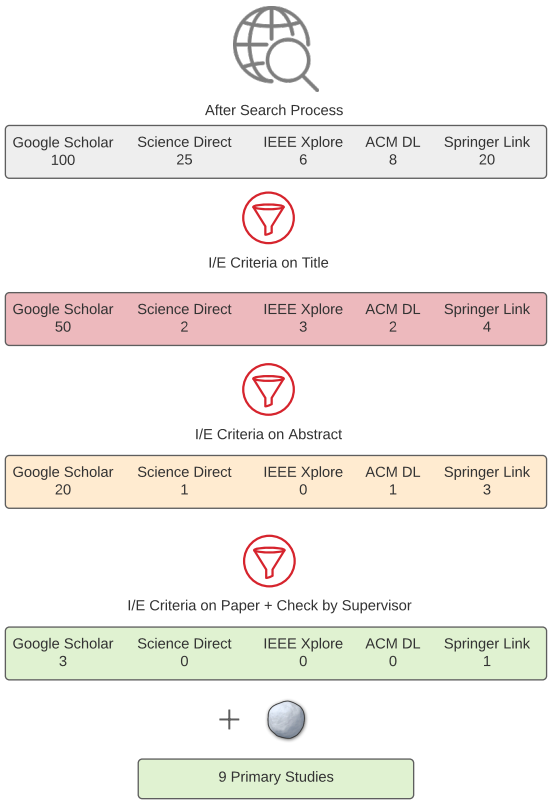

Figure 1: Amount of papers per stage in the review process

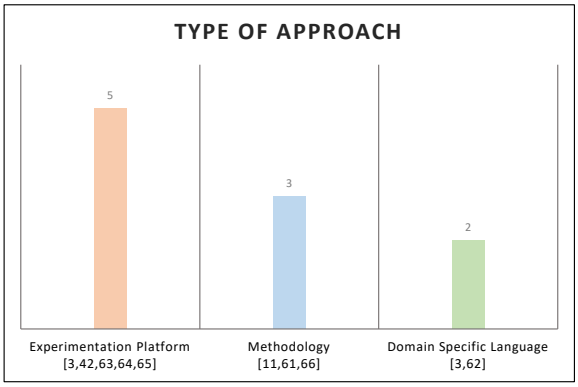

Figure 2: Bar chart showing the found approaches with the corresponding study ID at the bottom

are found and analyzed for their approach towards specifying A/B tests. Papers 3, 42, 63, 64, 65 describe experimentation platforms.

4.2.1 Similarities. The purpose of an experimentation platform is to run experiments more easily. Experimenters should be able to understand this platform and run their experiments. Therefore the users possibly do not have technical skills to manually create the experiment, thus as seen in the papers, the A/B tests are specified in a user interface. For example in PS 42, the experimenter has to make important choices when filling in the *audience*, *OEC*, *size and duration*, *experiment template*, *experiment interactions*, and *variant behavior*.

However, specification is only a part of an experimentation platform; developers also have to take other things into account e.g. structure within a company and how to analyze the results of A/B tests. Therefore the specification of A/B tests sometimes is not written as detailed as desired in these papers.

4.2.2 Differences. In PS 3 the experimentation platform of LinkedIn is demonstrated. The first thing an experimenter has to do is create a testKey that represents the feature or concept to be tested. After

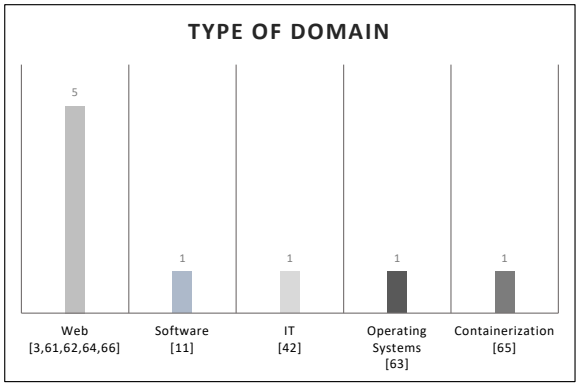

Figure 3: Bar chart showing the found application domains with the corresponding study ID at the bottom

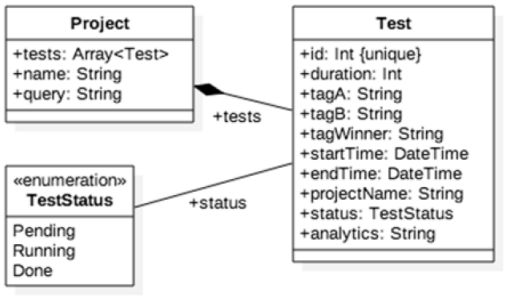

Figure 4: Class Diagram, from PS 65 [18]

the creation of the testKey, the A/B test is created as an instance of this testKey. With this approach, an A/B test is linked to a feature or concept and can be saved as so. PS 63 has a similar approach with a className and FeatureID shown at line 02 and 03 at the top of Figure 5. With this approach, the experiment is connected to a feature in the code using the className and FeatureID. However, there is no information about how the experiments are linked with e.g. a feature model. Moreover, in PS 65 a test is linked to a project, shown in Figure 4, but this perhaps is too broad when the total amount of tests begins to increase rapidly.

In PS 63, an experiment is defined using an XML format that is saved on the OS of a user, which can be run using a JSON payload (see Figure 5). The figure shows in the top part how an experiment is defined within an XML format. This includes *className, id, name, description* and the *variants*, which consists of an *id, enumName, name* and *description*. The bottom part of Figure 5 shows how an experiment can be run using a JSON payload. This payload contains a *version number, entityId, items to be tested (which includes featureId and variant),* and a *start* and *expire time*. This approach is done for the purpose of testing on the Windows operating system. In paper 64, the approach is explained rather informally compared to PS 63, by explaining the Optimizely UI. Figure 6 shows the User Interface (UI) of Optimizely. Using Optimizely, experimenters can run A/B/n

```
01: <feature>
02:   <className>
      Feature_StartSplitViewInMenuMode
      </className>
03:   <id>[FeatureId]</id>
04:   <name>
      Enable view selection in menu mode
      </name>
05:   <description>
        In desktop menu mode, Start now shows
        Either tiles or all apps instead of
        showing both at the same time
      </description>
06:   ...
07:   <variants>
08:     <variant>
09:       <id>1</id>
10:       <enumName>
          HideAppListOnByDefault
          </enumName>
11:       <name>
          Hide app list setting on by default
          </name>
12:       <description>
            Hide app list system setting for
            Showing the nav pane in menu mode
            is on by default
          </description>
13:     </variant>
14:   </variants>
15: </feature>
```

Figure 5. XML Experiment definition

```
01: {
02:   "v": "1.0",
03:   "ad": {
04:     "entityId": "[ExperimentIdentifier]",
05:     "items": [
06:       {
07:         "featureId": [FeatureId],
            "variant": 1
08:       }
09:     ]
10:   },
11:   "prm": {
12: "startTime":"[Start DateTime]",
    "expireTime":"[Expire DateTime]"
13:  }
```

Figure 9. Excerpt from assignment payload

Figure 5: Figures taken from PS 63 [16]

and multivariate tests on their websites. In the UI, experiments and audiences are split in different sections and where the experimenter can first create an audience and afterwards link that to an experiment. When segmenting an audience, the experimenter can set a multitude of attributes, e.g. accessing time of the website, browser type, localization, cookie information, if the user was funneled towards the website and even custom JavaScript functions. In addition, the audiences and experiments are defined in JSON configuration files (which are no examples of in the paper). We can speculate that these JSON configuration files will contain more

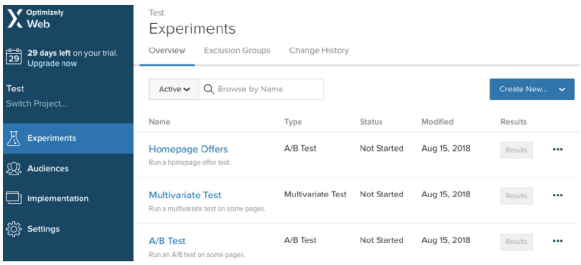

Figure 6: Optimizely UI, from PS 64 [11]

information as supposed to the JSON payload of PS 63, because both the audience and experiment are defined in it.

4.2.3 Opportunities. In PS 42, templates can be used to speed up the process of creating experiments. These templates are linked to different products, so when an experimenter is testing a certain functionality, it will be easier to set up the experiment. This isn't taken into account in other papers, conceivably a missed opportunity.

Including a clear relation between tests and earlier decided upon functionalities is shown in PS 63 and 65. In section 4.4 we will see a new approach that shows possible A/B tests in function requirements.

Lastly, in PS 3 a domain specific language is used. It will be interesting to see how that develops in the future, but the current developments are described next.

4.3 Domain Specific Language

Two papers are found that specify A/B tests by using a domain specific language (DSL). PS 3 is an experimentation platform where a DSL is used to save A/B tests, and PS 62 is a deep dive into a DSL called PlanOut.

An example of PS 3 is:

```
(ab (= (locale) "en_US")[treatment 10% control 90%])
```

An example of PS 62 is:

```
button_color = uniformChoice(
  choices=['#3c539a', '#5f9647', '#b33316'],
  unit=cookieid);
```

4.3.1 Similarities. It is unfortunate that no further examples are given in PS 3. In PS 3, experimenter first create the experiment by dividing the audience into segments to be able to target a specific audience. In the example of PS 3 above, the locale is set to English speakers in the US. However, other targeting methods are being used as well; for example *built-in Member Attributes* (e.g. country or last login-date), *Customized Member Attributes*, and *Real-time Attributes* (e.g. browser type or mobile device). In addition, experimenters can control how the audience is diverted over the variance. In the example this is a simple diversion of 10% to the treatment and 90% to the control group, but it is also possible to increase this 10% slowly and even expand the locale when a certain criteria is met. PS 62 shows a more detailed approach of using a DSL, but there are similarities between both papers. In PS 62, an experiment can also segment audiences using for example countries, but the amount of options to segment the audience seems lower than in PS 3. We will talk more about PS 62 in the next section.

4.3.2 Differences. The syntax of both approaches is different. An influence of this difference might be the fact that in PS 3 the A/B tests are stored using a DSL and in PS 62 the A/B tests can be run using a DSL. In addition, in PS 62 one can create A/B/n tests that uniformly direct the audience to different variants, however from reading the paper this does not seem possible in PS 3. Also, PS 62 shows a possibility to use conditional statements to be able to e.g. divide audiences and even run different experiments based on different conditions. In the example of PS 62 above,

4.3.3 Opportunities. In PS 3, observations can be made about which tests are executed in the past. But what about rerunning an experiment? Is it possible to use this language to run the experiment as well? This is not clear from the paper. Another interesting aspect of PS 3 is the possibility to ramp up the diversion of audiences and even expand the zone in which an experiment is run. This *dynamic* way of experimentation might be beneficial to PlanOut. In PS 62, the domain specific language is created in the environment of Facebook. Therefore it will be interesting to see how this can be generalized as much as possible to make this language useful for more companies.

4.4 Methodology

Three of the 9 papers are specified as using a methodological approach, which are PS 11, 61 and 66. The different specification approaches can be seen in Figures 7, 8 and 9.

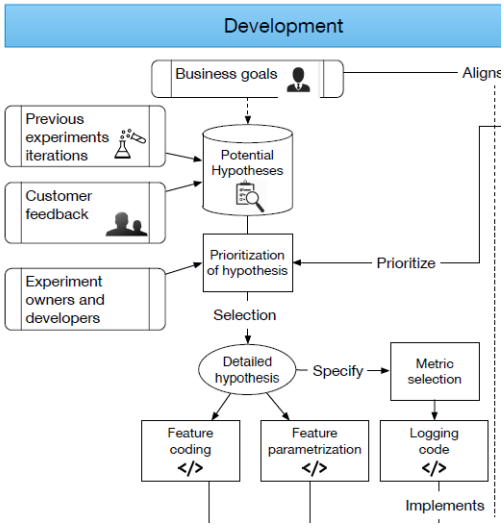

Figure 7: Part of Activity Model, from PS 11 [17]

4.4.1 Similarities. Comparing the taxonomy of an experiment in Figure 9 with the activities executed in the development process of an experiment in Figure 7, similarities are observed between what an experiment exists out of. For PS 11, especially the hypothesis part is important, because it will lead to the detailed hypothesis

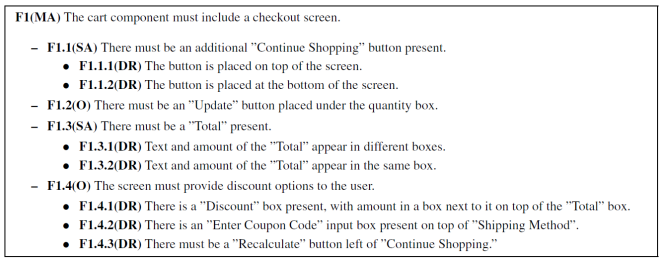

Figure 8: Feature Model, from PS 61 [4]

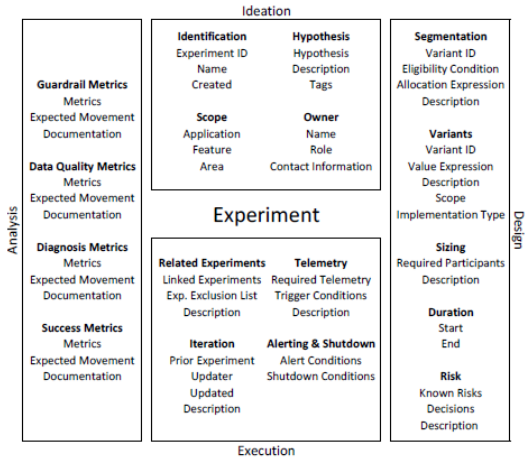

Figure 9: The experiment definition characteristics taxonomy, from PS 66 [1]

from which the metrics are selected and the implementation will be done, shown in the center of Figure 7. Moreover, part of the characteristics from PS 66 are in the detailed hypothesis of PS 11, including type of experiment, variants, segmentation, duration and metrics.

4.4.2 Differences. Looking at Figures 7, 8 and 9, all approaches are exceedingly different. PS 11 explains activities in the process of A/B testing, including part of the specification, which uses business goals, previous experiments iterations, customer feedback, and experiment owners and developers as input. This input is used to create a detailed hypothesis which in this case is the specification of an A/B test. PS 61 however, uses a feature model to specify part of A/B tests when creating features for a new or current part of the system. This is more aimed towards specifying the variants of A/B tests than actual A/B tests, because topics such as metrics and audiences are left out. Lastly, PS 66 describes a taxonomy of an experiment with all subjects that an experimenter has to keep in mind when doing an experiment.

4.4.3 Opportunities. The three approaches can be helpful for instance when creating an experimentation platform. The experiment definition characteristics taxonomy shows all the pieces that are needed in different phases to fully do an experiment. The activity model shows the order of activities of an experiment and what kind of inputs are required in order to specify the most fitting A/B test. Lastly, the feature model can be used when constructing features of a system, so that A/B tests can be specified more effortlessly by looking at the feature model for potential subjects for experiments.

5 DISCUSSION

As seen in Table 2, all papers are fairly new, which indicates that the sub-field of specifying A/B tests is in an early stage, but is beginning to get more attention. However, this literature review possibly has been done ahead of time; only 9 primary studies were found published between 2014 and 2020. The goal was to find 20 primary studies, so this could be a reason only 9 were found, but there can be other reasons for it as well (such as wrong search queries).

The results show a deep-dive into the different approaches of specifying A/B tests, including different experimentation platforms, domain specific languages and methodologies.

When the total amount of tests increases, keeping a clear overview can be harder. We have seen an approach that links the feature or concept to be tested to the actual test, but not in all approaches. This is an opportunity for future work in how to maintain overview when specifying A/B tests. One paper uses a feature model (Figure 8) where A/B tests are already specified within the feature specifications of a system.

Combining the results of this literature review to specify A/B tests can be valuable. Let us give some examples of how this can pan out. When creating an experimentation platform, the developer can use Figures 6 and 7 as a handhold to create the user interface and what inputs to use when specifying an A/B tests. In addition, using the experiment definition characteristics taxonomy in Figure 9 can help building the features of this experimentation platform. Looking at existing experimentation platforms, or using a DSL like PlanOut can give an advantage as well. When creating experiments in general, product owners can use the methods of feature models shown in Figure 8 to include potentially to be tested features when specifying features of a product.

6 THREATS TO VALIDITY

To describe the threats to validity accurately, we use the paper by Zhou et al. as a guide to structure the threats and to find possible threats being overlooked by us. [20]

6.1 External validity

In this literature review, there wasn't a focus on one particular application domain, but rather on multiple. However, most approaches found were in the Web domain. Therefore, generalizing this information over multiple application domains can lead to skewed results that are more aimed towards the Web domain. But we still chose for multiple application domains, because we found a low amount of primary studies in this field. Moreover, the papers describing experimentation platforms are written by well established and massive companies such as LinkedIn, Microsoft and Facebook. Generalizing this information allows these companies to gain ever more growth, while e.g. start-ups might be unable to grow from this information. Also, companies might not share specifications and instructions to their experimentation methods for confidential reasons and company reputation.

6.2 Internal validity

The search process was done by a master student without any "in the field" experience to conduct a professional literature review. First, the student had to learn the process to correctly do the literature review. In addition, the student had to deep-dive into the subject of A/B tests, because of a lack of knowledge. This lack of knowledge could lead to mistakes in the approach of the literature review, but also misclassifications of primary studies.

6.3 Construct validity

In this literature review there are potentially risks and biases, because the author was the only person analyzing papers. The author was supervised in the process by a professor from the Software and Sustainability research group of the Vrije Universiteit Amsterdam. The biases range from the social media usage of the author to the attitude towards doing user experiments. In addition, the selected results were only cross-checked by the supervisor, thus there was a shortage of different evaluations in this literature study as well.

6.4 Conclusion validity

Digital Libraries change over the years. The search queries could show different results the next day. Therefore all the found results to the search queries are saved and published on Github. In addition, the time stamp is added to the search results, so the reader knows in which time period the search has been done.

7 CONCLUSION

To answer the research question of this literature study, we can say that the current field of specifying A/B testing consists mainly out of 3 types of approaches: Experimentation Platforms, Domain Specific Languages, and Methodologies.

When creating an experimentation platform, a multitude of challenges arise, including specifying A/B tests. We have seen that it depends what the application domain is in which this experimentation platform is set up in; setting up experiments within operating systems have different challenges compared to their web counter parts. But there are similarities between them, which is helpful to focus on. Having a clear relation between a feature and an A/B test is shown to be helpful.

Using a domain specific language gives control to the developers what functionalities to include in this language, what to leave out, and how to make it easy to set-up A/B tests. Within Facebook, a DSL is created called PlanOut. However, this language is tested and created within Facebook's development environment. Applying this language in other environments is a next step in testing this language. Moreover, LinkedIn uses a DSL within an experimentation platform to store the specifications of an A/B tests, but it isn't clear whether it can be used to rerun experiments in their platform.

Methodologies define formal descriptions of approaches towards specifying A/B tests. The most promising paper found describes an experiment definition characteristics taxonomy. This taxonomy shows all parts that should be defined in order to create a good experiment. When creating an experimentation platform or specifying A/B tests, this taxonomy might be helpful to give developers a handhold when defining experiments in a system.

The field of specifying A/B tests is young but gaining more attention, since the found papers of this literature review span from 2009 until 2020. This literature review gives a grasp of what kind of different approaches exist towards specifying A/B testing. The results could help future developers that struggle with how to specify A/B tests, but also experimenters and researchers within the field of A/B testing.

REFERENCES

[1] F. Auer, C. S. Lee, and M. Felderer. 2020. Continuous Experiment Definition Characteristics. In *2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. 186–190. https://doi.org/10.1109/SEAA51224.2020.00041

[2] Eytan Bakshy, Dean Eckles, and Michael S. Bernstein. 2014. Designing and deploying online field experiments. In *Proceedings of the 23rd international conference on World wide web (WWW '14)*. Association for Computing Machinery, New York, NY, USA, 283–292. https://doi.org/10.1145/2566486.2567967

[3] Victor R Basili. 1992. *Software modeling and measurement: the Goal/Question/-Metric paradigm*. Technical Report.

[4] Javier Cámara and Alfred Kobsa. 2009. Facilitating Controlled Tests of Website Design Changes: A Systematic Approach. In *Web Engineering (Lecture Notes in Computer Science)*, Martin Gaedke, Michael Grossniklaus, and Oscar Díaz (Eds.). Springer, Berlin, Heidelberg, 370–378. https://doi.org/10.1007/978-3-642-02818-2_30

[5] P. Dmitriev, B. Frasca, S. Gupta, R. Kohavi, and G. Vaz. 2016. Pitfalls of long-term online controlled experiments. In *2016 IEEE International Conference on Big Data (Big Data)*. 1367–1376. https://doi.org/10.1109/BigData.2016.7840744

[6] Pavel Dmitriev, Somit Gupta, Dong Woo Kim, and Garnet Vaz. 2017. A dirty dozen: twelve common metric interpretation pitfalls in online controlled experiments. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1427–1436.

[7] Maria Esteller-Cucala, Vicenc Fernandez, and Diego Villuendas. 2020. Evaluating Personalization: The AB Testing Pitfalls Companies Might Not Be Aware of—A Spotlight on the Automotive Sector Websites. *Frontiers in Artificial Intelligence* 3 (2020), 20. https://doi.org/10.3389/frai.2020.00020

[8] R.A. Fisher. 1951. *The Design of Experiments*. Oliver and Boyd. https://books.google.nl/books?id=ejxBAAAAIAAJ

[9] Somit Gupta, Xiaolin Shi, Pavel Dmitriev, and Xin Fu. 2020. Challenges, Best Practices and Pitfalls in Evaluating Results of Online Controlled Experiments. In *Companion Proceedings of the Web Conference 2020 (WWW '20)*. Association for Computing Machinery, New York, NY, USA, 317–319. https://doi.org/10.1145/3366424.3383117

[10] Somit Gupta, Lucy Ulanova, Sumit Bhardwaj, Pavel Dmitriev, Paul Raff, and Aleksander Fabijan. 2018. The anatomy of a large-scale experimentation platform. In *2018 IEEE International Conference on Software Architecture (ICSA)*. IEEE, 1–109.

[11] Shan Jiang, John Martin, and Christo Wilson. 2019. Who's the Guinea Pig? Investigating Online A/B/n Tests in-the-Wild. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* '19)*. Association for Computing Machinery, New York, NY, USA, 201–210. https://doi.org/10.1145/3287560.3287565

[12] Kitchenham and Charters. 2007. Guidelines for performing Systematic Literature Reviews in Software Engineering. 2 (01 2007).

[13] Ron Kohavi, Alex Deng, Brian Frasca, Roger Longbotham, Toby Walker, and Ya Xu. 2012. Trustworthy online controlled experiments: Five puzzling outcomes explained. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 786–794.

[14] Ron Kohavi and Roger Longbotham. 2017. Online Controlled Experiments and A/B Testing. *Encyclopedia of machine learning and data mining* 7, 8 (2017), 922–929.

[15] Ron Kohavi, Roger Longbotham, Dan Sommerfield, and Randal M. Henne. 2009. Controlled experiments on the web: survey and practical guide. *Data Mining and Knowledge Discovery* 18, 1 (Feb. 2009), 140–181. https://doi.org/10.1007/s10618-008-0114-1

[16] P. L. Li, P. Dmitriev, H. M. Hu, X. Chai, Z. Dimov, B. Paddock, Y. Li, A. Kirshenbaum, I. Niculescu, and T. Thoresen. 2019. Experimentation in the Operating System: The Windows Experimentation Platform. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*. 21–30. https://doi.org/10.1109/ICSE-SEIP.2019.00011

[17] David Issa Mattos, Pavel Dmitriev, Aleksander Fabijan, Jan Bosch, and Helena Holmström Olsson. 2018. An activity and metric model for online controlled experiments. In *International Conference on Product-Focused Software Process Improvement*. Springer, 182–198.

[18] Ádám Révész and Norbert Pataki. 2019. Continuous A/B Testing in Containers. In *Proceedings of the 2019 2nd International Conference on Geoinformatics and Data Analysis (ICGDA 2019)*. Association for Computing Machinery, New York,

NY, USA, 11–14. https://doi.org/10.1145/3318236.3318254

[19] Ya Xu, Nanyu Chen, Addrian Fernandez, Omar Sinno, and Anmol Bhasin. 2015. From infrastructure to culture: A/B testing challenges in large scale social networks. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2227–2236. 2.

[20] X. Zhou, Y. Jin, H. Zhang, S. Li, and X. Huang. 2016. A Map of Threats to Validity of Systematic Literature Reviews in Software Engineering. In *2016 23rd Asia-Pacific Software Engineering Conference (APSEC)*. 153–160. https://doi.org/10.1109/APSEC.2016.031