Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 1115 – 1128

doi:10.3906/elk-1610-50

Usability evaluation of a web-based ontology browser: the case of TSONT

Türkan KARAKUŞ YILMAZ[1,*], Umut DURAK[2], Halit OĞUZTÜZÜN[3], Kürşat ÇAGILTAY[4]
[1]Department of Computer Education and Instructional Technology, Kazım Karabekir Faculty of Education, Atatürk University, Erzurum, Turkey
[2]German Aerospace Center, Braunschweig, Germany
[3]Department of Computer Engineering, Faculty of Engineering, Middle East Technical University, Ankara, Turkey
[4]Department of Computer Education and Instructional Technology, Faculty of Education, Middle East Technical University, Ankara, Turkey

Received: 06.10.2016 • **Accepted/Published Online:** 06.01.2018 • **Final Version:** 30.03.2018

Abstract: As the use of ontologies expands, their visualization is becoming increasingly important. In this study, an ontology browser for visualizing the Trajectory Simulation ONTology (TSONT) was evaluated in terms of usability by considering its subdimensions, which are effectiveness, efficiency, and user satisfaction. The methodology employed in this study for evaluating an ontology browser is reported along with the results of the evaluation. The TSONT browser is a tree-type ontology browser created to allow developers to visualize TSONT. Six flight simulation programmers with at least one year of experience participated in the study. The participants were given usability tasks and their voices and eye movements were recorded using a sound recorder and eye-tracker, respectively. The results not only showed that guidance and terminology influence the efficiency, effectiveness, and user satisfaction of the ontology browser, but they also revealed important insights into the requisites of the general usability of ontologies, even in simple text-based interfaces.

Key words: Web-based ontology browsers, ontology visualization, usability, trajectory simulations

1. Introduction

Ontology as an area of philosophy means "knowledge of existence." Gruber [1] defined it as the explicit description of a conceptualization. An ontology may take a variety of forms, but it will necessarily include a vocabulary of specific meaning. This vocabulary includes definitions and indications of how concepts are interrelated, which, collectively, imposes a structure on the domain and constrains possible interpretations of the terms [2]. An ontology includes taxonomies, which are widely used to capture generalizations, and specializations related to a domain [3]. Knowledge in ontology is formalized using five types of components: concepts, relations, functions, axioms, and instances [1]. For database systems, ontologies might mean levels of abstraction in data models, analogous to hierarchical and relational models that display knowledge about individuals, their attributes, and their relationships to others [4].

Ontologies are key for successful knowledge sharing in engineering. They enable knowledge reuse and the standardization of terminology [4]. The Trajectory Simulation ONTology (TSONT) was developed to be a domain model of simulation infrastructure that allows reuse of domain knowledge to code implementation [5,6]. The structure of TSONT has been devised to render concept-to-implementation mapping amenable to reuse by

*Correspondence: turkan.karakus@gmail.com

trajectory simulation developers. The simulation is formalized as a subentity of the Trajectory Simulation Class and defined by number of properties, such as hasPhase, hasTrajectory, and servesComputeTrajectory. These properties constitute the definition of trajectory simulation. The formalized domain knowledge in TSONT is considered useful both (a) in model-driven engineering approaches to simulation designs by utilizing machine processing [6] and (b) in sharing the engineering knowledge used to construct trajectory simulations with developers.

Ontologies have been used by researchers and application developers in many domains to solve a wide range of problems, including data integration, configuration, data analysis, annotation, and search [7,8]. TSONT is planned to be sustainable and developable as the infrastructure reuse continues. With the experience gained by new projects, it will become more mature and complete. Protégé is used as the ontology development environment and enables an integrated formalization of the captured conceptualization, while constructing a visual representation of the ontology [9].

As a web-based ontology browser, TSONT is also intended to provide easy, reusable, and shareable ontological data [10]. Since the development of ontologies, the literature has mostly focused on design recommendations [11–13]. As [10] has indicated, if web ontologies provide easy use of ontological data, then it might be helpful for other users to utilize, expand, and interconnect them. However, formal evaluations of many ontologies have shown that they neither provide ease-of-use when dealing with multiple ontologies nor enhance reusability [14]. Moreover, no general agreement has been reached on an effective structure by which to utilize ontologies to secure improved knowledge management and decision making [15]. Thus, this study aims to evaluate a web-based ontology browser to improve its design and to exemplify a model for ontology browsers by applying a user-centered approach. This approach considers users who may be in different settings using different applications.

Usability is an umbrella term composed of effectiveness, efficiency, and user satisfaction toward any tool. A usability evaluation considering efficiency, effectiveness, and user satisfaction is based on user experience, which means that those factors can be measured by considering user actions and attitudes. This study is important in regard to exemplifying how a user-centered approach can be used in ontology browser evaluation. Further, it applies an original quantified approach in order to prove interface efficiency and demonstrate how quantitative and qualitative data can be combined to determine the usability of ontology browsers. In this article, ontology and usability evaluations are first mentioned and a method is drawn. The positive and negative aspects of the usability issues, according to the usability test results, are reported and then discussed.

1.1. Evaluation of ontologies

Effective ontologies require a well-designed and well-defined ontology language [16]. In addition, reasoning is an important aspect to ensure that concepts are noncontradictory, related, and ordered in a correct hierarchy. This organization of concepts will facilitate the rich structure of ontologies and ontology-based information [16]. Ontology evaluation generally combines a verification and validation process [17]. A verification process assesses whether the ontology building function is suitable for the real world. In a validation process, on the other hand, the meaning of ontology is evaluated in order to check whether the definitions model the real world. Apart from validation and verification, a user-based assessment is used as a third option in order to understand the usability, usefulness, and portability of the ontology in regard to the user's point-of-view [17].

As [18] suggested, the evaluation of an ontology must consider its usability, usefulness, abstraction levels, quality, granularity, and portability of concept. According to [19], in order to gain a consistent level of quality and consequent acceptance of industry, ontology-based tools must be evaluated in terms of usability. Therefore, this study attempts to conduct a usability evaluation of an ontology tool in order to exemplify how to test the usability of an ontology quantitatively and evaluate it qualitatively.

Usability is pivotal for ontology browsers to satisfy user expectations. Satisfaction has been defined as "the extent to which a product can be used by specified users to achieve specified goals with *effectiveness*, *efficiency*, and *satisfaction* in a specified context of use," according to ISO 9241-11, 1998 standards [20]. Therefore, our evaluation was composed of those three factors. Each of these factors has many dimensions. In order to determine the usability dimensions to be measured, the purpose of the interface must be taken as a base [21]. Reference [22] identified several website usability dimensions to consider: consistency, navigability, supportability, learnability, simplicity, interactivity, telepresence, credibility, readability, and content relevance. However, these dimensions might be considered too broad for simple interfaces.

The effectiveness of a usability goal is mostly related to its functionality [23]. The effectiveness of an ontology is likely to increase with feedback after consecutive evaluations [24]. An effective ontology should provide clarity (i.e. objectives and complete definitions of the concepts), coherence (i.e. consistent logic), extendibility (i.e. allowing for new concepts with no need to change available concepts), minimal encoding bias (i.e. concepts should not be context-dependent and should allow for knowledge sharing without bias), and minimal ontological commitment (i.e. using its weakest theory to make useful claims about the real world) [17,24,25]. Reference [18] proposed very similar criteria to evaluate ontologies, such as consistency, completeness, conciseness, expandability, and sensitivity.

In user-centered usability measures, time to learn, speed of performance, and rate of error are strong metrics to assess the efficiency of an interface [26]. Related learnability and navigability dimensions are also important elements [21]. Since subjective satisfaction is related to how much users are pleased with using an interface, the impression of users and how much they want to use the interface again would be useful information to decide on the satisfaction.

User-centered evaluations are becoming one of the most popular methods of ontology evaluation [27–29]. Evaluations conducted with target users can verify architecture, content, syntax, and the software itself [24]. Reference [24] suggested that new studies must collaborate with users in order to identify the essential characteristics of ontologies. Usability studies of ontology browsers should also be conducted to reveal why users fail at certain tasks and what can be done to encourage users to complete tasks [30]. User-centered evaluations are particularly important for novice users who need effective tool support to understand the content and structure of ontologies [31]. Due to the aforementioned advantages of user-centered designs, this study embraced a user-centered approach to evaluate the effectiveness, efficiency, and user satisfaction of an ontology browser. For the sake of efficiency, time to learn, speed of performance, and error rates were evaluated quantitatively. For the sake of effectiveness, a more qualitative approach was used to reveal clarity, coherence, extendibility, minimal encoding bias, and ontological commitment of ontology. Finally, an overall evaluation of users was taken to evaluate user satisfaction.

1.2. The TSONT browser case study

Trajectory simulations compute flight paths as well as other values of interest related to the motion of air vehicles. The purpose of trajectory simulations is to provide users with data in order to better understand the air vehicle for a variety of tasks, such as performance requirements, designing, optimizing design parameters, and training. The TSONT Browser is a web-based ontology browser (see Figure 1). It provides a basis for specifying the requirements of trajectory simulation applications and is being elaborated upon as a reusable trajectory simulation knowledge library for developers. After considering the structure of TSONT [5], the tree view was selected from among the four main ontology representation schemas over network view, neighborhood view, and hyperbolic view [26].

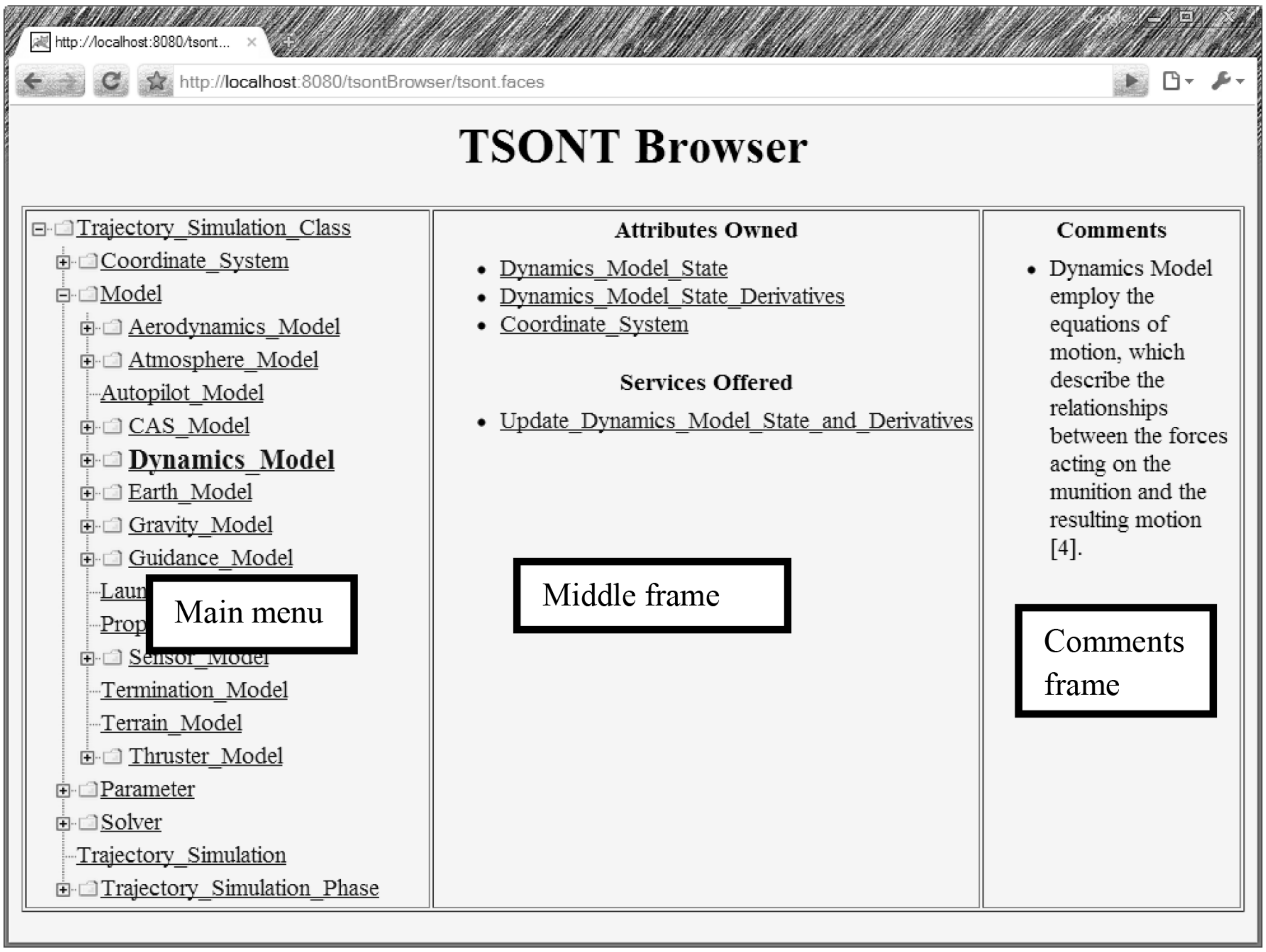

Figure 1. TSONT browser interface.

The TSONT Browser parses the OWL ontology and serves the content in a container. As depicted in Figure 1, the classes of the ontology are presented at the left in a hierarchical manner. The middle pane lists the attributes of the selected class, while the right pane shows the annotations of the selected class. The TSONT ontology web browser is composed of a main tree structure with subnodes and incidental links. After selecting a main node, the links appear in the middle under Attributes Owned and Services Offered. Comments appear on the right frame, if available.

Figure 2 presents the links and subpages. Users select a main node to see its attributes; upon clicking a node, its text becomes bold. If a participant clicks a link in the middle frame, then a new subpage opens and new links appear under Inputs, Outputs, Dependencies, or Hierarchy. If the participant clicks on a link on the first subpage, another subpage appears displaying Records. Titles for the same level of subpages are not fixed and change in accordance with the attributes of the upper nodes.

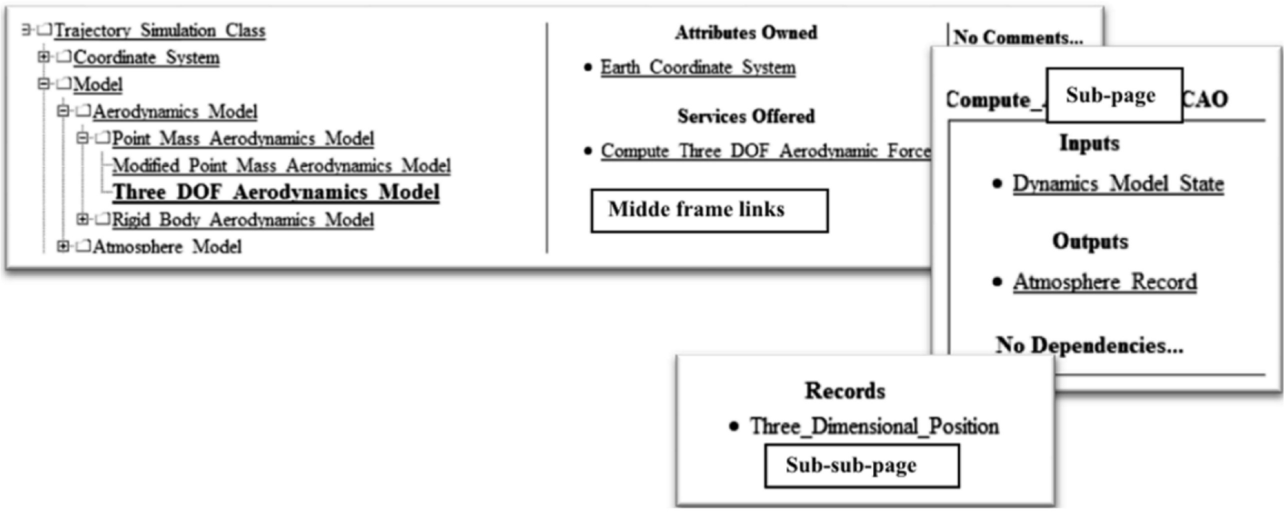

Figure 2. Subpages of TSONT browser.

The ontology browser has been developed and used for various occasions, particularly for TSONT. While there have been a number of publications that depict excerpts from TSONT, due to confidentiality concerns, the overall ontology is never disclosed in a public URL.

2. Materials and methods

2.1. Preanalysis and instruments

In the preanalysis stage, a draft task analysis of the TSONT browser was performed. Accordingly, the researchers determined the tasks and questions to be asked before and after implementation in order to assess browser usability. Related tasks were created in order to emphasize effectiveness [3] and efficiency [32]. Based on [26]'s usability metrics, [32]'s task creation standards included overview (i.e. a heuristic evaluation of the whole system), zoom (i.e. finding intended items by zooming), filter (i.e. hiding uninteresting items), details-on-demand (i.e. viewing classes and their properties), relate (i.e. view relationships among items), and history (i.e. a history of actions to support undo and replay). These actions were also considered when developing the tasks.

Four question categories emerged to appraise the satisfaction of the users: preknowledge and demographic questions, tasks for efficiency, questions about effectiveness, and post questions for satisfaction. Applying the think aloud process while administering the usability instrument provided clues about the effectiveness and efficiency of the browser.

A pilot study was conducted with one trajectory simulation developer and, from the results, the clarity of the tasks was improved. At the end of the pilot study, the researchers decided to ask the effectiveness questions in accordance with the users' actions in the browser. Asking about predetermined words might not make sense for the participants and so it was decided that the participants would be asked about random words on the screen, reasons for any choices, and whether they understood the concepts encountered. In addition, the participants were asked what they thought about denominations and whether they were familiar with these concepts.

For usability purposes, the researchers covered several types of information searches. Names, functions, definitions, and attributes were asked about in different forms to reveal challenges with the hierarchy or names of the concepts. Error rates, the numbers of steps necessary to achieve tasks, time to learn, and speed of performance gave additional clues about the comprehensibility of the browser. Table 1 shows the efficiency tasks and the minimum number of steps required to achieve them.

Table 1. Efficiency measurement tasks and number of steps.

Task	Number of steps to achieve it
By making use of TSONT, answer what constitutes state and state derivates of the dynamic model that you use in a point mass trajectory simulation within three degrees of freedom.	6 Steps
By making use of TSONT, give a name to the model that computes aerodynamic forces and moments in a trajectory simulation within six degrees of freedom.	4 Steps
What constitutes the structure that defines acceleration vector in TSONT?	5 Steps
What is the responsibility of the Coordinate System class in TSONT?	2 Steps
How are the physical characteristics of vehicles represented in TSONT for point mass simulations?	5 Steps

The pilot study tasks were also evaluated in terms of comprehensiveness. A combination of the tasks required the users to check all of the windows of the web browser. In order to assess effectiveness, the researchers asked questions about concepts in the ontology corresponding to certain goals (see Table 2).

As seen in Table 2, all of the criteria were covered by the tasks. The researchers changed the concepts presented in the tasks in accordance with the performance of the participants.

Table 2. Effectiveness measurement tasks.

Task	Goal
Participants were given a concept by asking whether they understood what its function was in the browser	Clarity
Asked whether the name of the function was coherent with its function	Coherence
Asked whether the concepts given in TSONT were familiar to them	Extendibility
Asked whether any concepts existed that they perceived in a different way	Minimal encoding bias
Asked whether TSONT seemed suitable to standards	Minimal ontological commitment

2.2. Sample

The evaluation was carried out with six trajectory simulation programmers whose ages spanned 24 to 29. Three of the participants had aerospace engineering backgrounds, two were from mechanical engineering, and the last one was from computer engineering. These participants were highly experienced in aerospace engineering and knowledgeable about the functions of ontologies, but did not have any experience with web-based ontologies or TSONT. Their trajectory simulation experience ranged from one to seven years, with an average of two and a half years. They were specialists at companies in the aerospace industry.

User-centered studies are commonly conducted with five or more participants. In fact, four to five participants uncover 80% of usability problems [33,34]. After the fifth participant, more participants might uncover much fewer new usability problems [34,35]. In this study, there were seven potential participants, but one left the pilot study. Therefore, we had six participants. As the target population for the study was very limited, the six participants were enough to determine the usability of the web browser, which was specialized for a task.

2.3. Data collection procedure

The participants were invited to the Human Computer Interaction Laboratory (HCI Lab - http://hci.cc.metu.edu.tr/en/) in order to record their gazing data by means of an eye-tracking device. The HCI lab is located in the Informatics Unit on the Middle East Technical University campus, which is located in Ankara. It was predicted that each session would take about half an hour for each user, but there was no time limitation while conducting the tasks. The efficiency and effectiveness tasks were given to the participants and a think aloud procedure was applied. The tasks were presented to the participants sequentially (i.e. once a task finished, the next task was given). While the participants completed the tasks, the researchers documented the video recording via an eye-tracking device, documented the sounds by means of a sound recorder, and noted all of the participants' behavioral actions. It should be noted that fixation durations and numbers were not used in the analysis, since the pages were dynamic and each user used the same or different links, went forward and back, opened different pop-up windows, and slide or scrolled through these windows. These variables made it difficult to calculate the fixation measures for a specific window. Therefore, in this study, the eye-tracking data were used for the timing calculations and make some qualitative interpretations via videos. After completing all of the tasks, participants were asked to demonstrate their level of satisfaction with a 7-point scale, with 7 meaning "very easy."

2.4. Data analysis and reporting procedure

While creating the usability evaluation criteria, [26] four measurements of human factors—time to learn, speed of performance, rate of errors, and subjective satisfaction—were utilized. These criteria were also taken into consideration while making user satisfaction evaluations. The screen records of the eye-tracking data allowed the researchers to evaluate the performances qualitatively.

3. Results

The findings of the study will be handled under the three dimensions of usability.

3.1. Efficiency of the browser

In order to define the efficiency of the browser, three criteria were used: time to learn, speed of performance, and error rate [26].

3.1.1. Time to learn

The tasks were applied in the same order given in Table 1. The time consumed by the users for each task was recorded in order to compare the tasks and determine whether the participants used less or more time after gaining familiarity with the browser. Based on the eye-tracking data, the users tended to begin with a glance at the whole structure of the tree (see Figure 3).

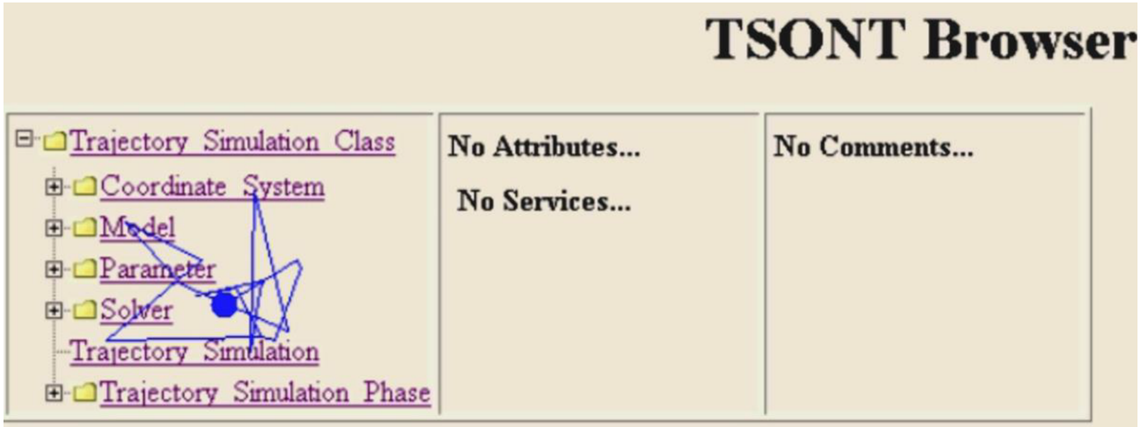

Figure 3. Eye movements within the TSONT browser.

The first two tasks were related; therefore, the second task may naturally have taken less time. One participant could not understand what he was supposed to do and so he tried several different ways to find an answer and increased the average time for Task 2. The first four tasks included a keyword like "model" or "solver" that gave a clue about which node to select. The fifth task had no clue word, but was based on the previous experiences and expectations of the participants. Although the participants were asked for a "parameter," they tended to look at the model node first, searching for a special model. Two of the participants tried to find "physicals" in the data nodes. Ultimately, 83% of the participants tried a node other than "parameter" in their first attempts.

In order to define the time-to-learn variable, the researchers subtracted the duration of a task from the duration of the following task for each participant and then calculated the average duration difference, except for Task 5. As seen in Table 3, the duration for the consecutive tasks reduced dramatically.

Table 3. Differences between task-to-task durations.

Tasks	Average difference (s)	Difference (average diff/ previous task duration) (%)
Task 2 – Task 1	17.3	13.3%
Task 3 – Task 2	36.5	32.5%
Task 4 – Task 3	36.5	48.3%

This growing reduction in duration may signify a reasonable ability to acclimate to this browser, especially for users with experience in trajectory simulations.

3.1.2. Speed of performance

Another efficiency clue is speed of performance, which is indicated by the average time taken for each task. The users tended to access the middle frame in order to find the features or functions of the models. In this study, the users had unlimited time and so to define speed performance the number of required steps to achieve the tasks and the number of steps taken by each user were compared. The average time and average number of steps for each task offered information about speed performance. Each task was expected to take two to six steps. As seen in Table 4, the users often took twice as many steps as needed for each task. In some cases, the users did not recognize that they had already found the answer and so they continued exploring, increasing the time expended.

Table 4. Task completion times and steps.

Tasks	Average duration (s)	Average number of steps	Fastest time (s)	Slowest time (s)
Task 1	129.3	12	70	210
Task 2	112.0	7.8	80	170
Task 3	75.5	7.2	23	135
Task 4	39.0	3.2	12	95
Task 5	136.3	13.2	34	255
Average	98.4	8.7	43.8	173

The average time for total task completion was 492.2 s and the average number of steps taken for each task was 8.7. Table 3 shows a tremendous difference between the fastest and slowest times. One explanation is that experienced users completed the tasks very quickly. Plus, due to a lack of prior knowledge and insufficient guidance in the browser, not all of the participants recognized the correct answers. Some of the participants tried to answer using their background knowledge when challenged by the browser.

3.1.3. Error rate

The numbers of errors and completed tasks provided indications about the efficiency of the design as well. The error rate was not high: the total number of tasks was 30 (six users $\times$ five tasks) and only 10% of the tasks were left incomplete. However, for 28.6% of the tasks, the participants did not initially realize that they had found the answer (see Table 5).

Table 5. Types and numbers of errors.

Tasks	Incomplete task	Beginning with wrong link	Ignoring right answer	Giving wrong answer	Total
1	1	3	1	2	7 (33.3%)
2	-	2	3	1	6 (28.6%)
3	-	2	1	-	3 (14.3%)
4	-	-	-	-	0 (0%)
5	1	2	1	1	5 (23.8%)
Total	2 (9.5%)	9 (42.8%)	6 (28.6%)	4 (19%)	21 (100%)

As seen in Table 5, most of the errors were caused by flawed first attempts. As stated above, some of the participants ignored the right answer because of assumed prior knowledge or a lack of guidance. If a participant provided a wrong answer, then the researchers assigned the task again. Out of the 261 steps matched to complete tasks (six participants $\times$ five tasks), 7% of the steps had errors. This rate can be accepted as low because it excluded incomplete attempts.

3.2. Effectiveness – ontological perspective

Qualitative data were used to interpret those factors influencing the effectiveness of the ontology browser.

3.2.1. Clarity and coherence

Clarity and coherence appear together because they overlap in this context. These two factors were qualitatively interpreted from data that included the participants' views on several ontological concepts as well as vocabulary consistency and functions of the nodes. In order to define clarity and coherence, all of the participants were randomly asked about denominations in the middle frame and on other subpages. In three cases, the participants stated that they did not know the function of a link, but could predict it. For example, the researchers asked a participant to guess the function of Dependencies, a title on a subpage, and he stated "actually, it does not give any clue about its function. I should click on links under this title to understand what I can do by using this part. It might be a thing related to what solver is influenced." This expression gave a clue about the trajectory simulation experience and was very effective with regard to browser clarity, since it was specific to the field; knowledgeable engineers were comfortable predicting the functions of each title.

In the fifth task, the participants were asked about the physical properties of a trajectory simulation. Two of the participants asked whether the question was for a specific model because physical characteristics tend to vary. Although they were seeking "features," the answer was under "records" due to some automatic and programmed denominations. Each case might need a specific denomination; therefore, developers should plan for titles and subpages to provide coherence. These sections could benefit from more comprehensible titles. In a task completion session, two of the participants asked which model they were supposed to find the physical properties for in a trajectory simulation. There was only one "physicals" title in the "parameters" node and the participants were supposed to use the "physicals" link, but they could reach the same page two other ways.

3.2.2. Extendibility

Four of the participants stated that they found the denomination to be meaningful, reasonable, and understandable. One participant stated that "several other resources give different names to models or classes; therefore, this should be considered while designing this browser." The extendibility of this browser is clear because it has a very basic structure and new denominations and links can be added easily. The browser also uses different library files, which means that, even without manipulating the main code, other files can be modified. The files are divided by function and so the programmers can easily access and manipulate only the relevant files.

3.2.3. Minimal encoding bias

Minimal encoding bias can be prevented by making a tool context-independent. However, this tool was developed specifically as a guide for the trajectory simulation developers. An ontology presents many useful models and all classes of trajectory simulation. Context dependency offers experienced users good interactions and high comfort, but less experienced users face challenges since they have to apply trial-and-error techniques to find the correct information. Context dependency does not prevent functionality related to this browser because the users recognize that information is for a particular job and will easily accomplish trajectory simulation tasks. However, this case does not mean that the browser cannot be extended; its code structure can be manipulated and other tree structures can be added for different simulations.

3.2.4. Minimal ontological commitment

In order to provide flexibility, a browser should use the simplest ontological structure. The ontology browser relies on a real structure of trajectory simulation and presents all required items with real world definitions. It provides properties, services, inputs, and outputs for almost all of the trajectory simulation models, allowing users to make choices from a wide spectrum of information. This flexibility reduces the browser's ontological commitment, since it aims to serve all of the needs of the simulation code developers. On the other hand, this large range of information could be designated in a simpler manner. As stated above, in some cases, no consistency exists between the information given in the middle frame and on the other pages. For example, the parameters and models could be integrated because the users tend to think that the parameters are part of the model.

3.3. Subjective satisfaction

After completing all of the tasks, the participants were asked to rate the difficulty of the browser on a scale from 1 to 7 (7 = very easy). The average score was 5.5. Although the score implies a possibility of ease-of-use, it is difficult to draw a generalization because of the number of participants. It is important to note that, during the think-aloud procedure, most of the participants expressed issues pointing to complex or vague instructions. A total of 13 negative and 14 positive views were stated by the users. All of the participants agreed that, after a little practice, the interface was very easy to use, but they complained about a lack of guidance, the need for trial-and-error practice, unclear denominations (especially for novices), a lack of information about interface motivations, and visibility issues when nodes opened and other frames disappeared.

While the positive and negative statements about the interface were roughly equal, the negative statements were mostly related to learnability (n = 10, 13 total issues) and the positive statements were related to the efficiency and effectiveness (n = 10, 14 total issues) of the interface (see Table 6). Although learnability was the most negative issue according to the participants, the results related to time to learn and speed of performance showed that it was easy to become familiar with the browser.

Table 6. Users' views on usability patterns.

Usability pattern	Negative	Positive	Total
Learnability	10	4	14
Effectiveness	1	3	4
Efficiency	2	7	9
Total	13	14	27

The expectations of the experienced engineers primarily led to the negative views. When evaluating the browser, they mostly stated that it would be difficult for novices to use and emphasized how they did not want to apply trial-and-error methods, even if they took less time. The participants had fewer issues regarding effectiveness than they did with efficiency and learnability. This finding may be due to a low incomplete task rate.

4. Discussion and conclusion

This study revealed a method used to evaluate three usability issues for a trajectory simulation ontology browser: efficiency, effectiveness, and user satisfaction. Eye-tracking provided data for the timing and, by using this information, the researchers developed new strategies to measure efficiency [36] and enhanced the study with qualitative and quantitative data [37].

The participants managed to complete almost all of the tasks without intervention, or through a combination of trial-and-error and reminders. Speed of performance was mostly influenced by prior knowledge and an understanding of what was expected by the task. Therefore, the assigned tasks should be defined in several steps in order to ensure maximum clarity. Despite revisions after the pilot study, some of the users did not always understand what they were supposed to do and sought clarification several times. This difficulty might have been caused by a lack of documentation on the purpose of the browser and how it would be used [38] as well as incomplete previous knowledge.

The efficiency results were mixed: two of the tasks were abandoned before completion, but the participants used less time on tasks following those tasks, despite having no prior experience with the ontology browser. Learnability was met when considering time to learn and achieved tasks, but the engineers expressed many issues that might complicate browser ergonomics. All of the participants had computer-related jobs and actively used computers and the Internet; therefore, they were already familiar with this type of browser. A simplistic structure, strong categorization, and predictable link names might have affected these results [38]. On the other hand, although the quantitative data provided good insights for usability, the qualitative evaluation showed room for improvement. For example, clearer categorizations could be made, such as in the "parameter" node, and the comments could be used more actively with a better classification system. In fact, quantitative measures were not enough to make generalizations with 6 participants. Although six participants were enough to reveal as much usability problems, increasing the number of participants might provide more reliable measures for some of the variables, such as time to learn, error rate, and ease-of-use.

This paper exemplified an evaluation of a newly developed ontology browser and provided suggestions to improve its usability. The findings will be especially helpful for engineers developing ontology browsers. The results showed how experienced engineers can face challenges in simple interfaces and how even the simplest browsers have many usability issues. This paper also emphasized the awareness of usability requirements for a simple interface. A collaborative and participatory design approach might be adapted going forward to develop more usable ontologies [39]. This method requires user involvement during each step of ontology development. Thus, designers go beyond usability to provide user satisfaction based on preferences. For ontology browsers, confusion related to links in the same node should be clarified by simplifying content and explaining the processes to users. The extendibility of the browser can easily accommodate these changes to the content and denominations of the ontology browser [40].

References

[1] Gruber TR. Towards principles for the design of ontologies used for knowledge sharing. Int J Hum-Comput St 1995; 43: 907-928.

[2] Uschold M, King M. Towards a methodology for building ontologies. In: Workshop on Basic Ontological Issues in Knowledge Sharing. Montreal, Quebec, Canada, 1995.

[3] Durak U, Schmidt A, Pawletta T. Ontology for objective flight simulator fidelity evaluation. Simulation Notes Europe, 2014; 24: 69-78.

[4] Mizoguchi R, Ikeda M. Towards ontology engineering. Journal-Japanese Society for Artificial Intelligence. 1998; 13: 9-10.

[5] Durak U, Oguztuzun H, Ider SK. An ontology for trajectory simulation. In: Proceedings of the 2006 Winter Simulation Conference. pp. 1160-1167.

[6] Durak U, Oguztuzun H, Ider SK. Ontology based domain engineering for trajectory simulation reuse. Int J Softw Eng Know 2009; 19.

[7] Tudorache T, Nyulas C, Noy NF, Musen MA. WebProtégé: A collaborative ontology editor and knowledge acquisition tool for the web. Semant Web 2013; 4: 89-99.

[8] Lohmann S, Link V, Marbach E, Negru S. WebVOWL: Web-based visualization of ontologies. In: Lambrix P et al., editors. Knowledge Engineering and Knowledge Management. EKAW 2014. Lecture Notes in Computer Science, vol 8982. Cham, Switzerland: Springer, 2015. pp. 154-158.

[9] Peroni S, Shotton D, Vitali F. Tools for the automatic generation of ontology documentation: A task-based evaluation. Int J Semant Web Inform Syst (IJSWIS) 2013; 9: 21-44.

[10] Kalyanpur A, Parsia B, Sirin E, Grau BC, Hendler J. Swoop: A web ontology editing browser. J Web Semant 2006; 4: 144-153.

[11] Gangemi A, Presutti V, Blomqvist E. The computational ontology perspective: Design patterns for web ontologies. In: Sartor G, Casanovas P, Biasiotti M, Fernandez-Barrera M, editors. Approaches to Legal Ontologies: Theories, Domains, Methodologies. Law, Governance and Technology Series. Dordrecht, the Netherlands: Springer, 2011. pp. 201-217.

[12] Hervás R, Bravo J, Fontecha J. A context model based on ontological languages: A proposal for information visualization. J Univers Comput Sci 2010; 16: 1539-1555.

[13] Uschold M, Gruninger M. Ontologies: Principles, methods and applications. Knowl Eng Rev 1996; 11: 93-136.

[14] Peroni S, Motta E, d'Aquin M. Identifying key concepts in an ontology, through the integration of cognitive principles with statistical and topological measures. In: Proceedings of ASWC 2008; 8–11 December 2008; Bangkok, Thailand: Berlin, Germany: Springer. pp. 242-256.

[15] Haghighi PD, Burstein F, Zaslavsky A, Arbon P. Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings. Decis Support Syst 2013; 54: 1192-1204.

[16] Baader F, Horrocks I, Sattler U. Description Logics. In: Staab S, Studer R, editors. Handbook on Ontologies. Berlin, Germany: Springer, 2013. pp. 3-28.

[17] Gomes-Perez A. Ontology evaluation. In: Staab S, Studer R, editors. Handbook on Ontologies. Berlin, Germany: Springer, 2013. pp. 251-274.

[18] Gómez-Pérez A. Evaluation of ontologies. Int J Intell Syst 2001; 16: 391-409.

[19] Giboin A, Gandon F, Corby O, Dieng R. Assessment of ontology-based tools: a step towards systemizing the scenario approach. In: EON2002: Evaluation of Ontology-Based Tools Workshop at the 13th International Conference on Knowledge Engineering and Knowledge Management EKAW 2002; 30 September 2002. pp. 63-73.

[20] ISO 9240-11 (1998). Ergonomic requirements for office work with visual display terminals (vdts) part 11: Guidance on usability.

[21] Albert W, Tullis T. Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics. Waltham, MA, USA: Elsevier, 2013.

[22] Lee Y, Kozar KA. Understanding of website usability: specifying and measuring constructs and their relationships. Decis Support Syst 2012; 52: 450-463.

[23] Shiffman RN, Liaw Y, Brandt CA, Corb GJ. Computer-based guideline implementation systems. J Am Med Inform Assn 1999; 6: 104-114.

[24] Gomez-Perez A, Juristo N, Pazos J. Evaluation and assessment of knowledge sharing technology. In Mars NJ, editor. Towards Very Large Knowledge Bases – Knowledge Building and Knowledge Sharing Technology. Amsterdam, the Netherlands: IOS Press, 1995. pp. 289-296.

[25] Yu J, Thom JA, Tam A. Ontology evaluation using Wikipedia categories for browsing. In Proceedings of CIKM 2007, 6–10 November 2007; Lisbon, Portugal: ACM. pp. 223-232.

[26] Shneiderman B. Designing the User Interface. Boston, MA, USA: Addison-Wesley, 1996.

[27] Forsell C, Cooper MC. An introduction and guide to evaluation of visualization techniques through user studies. In: Huang W, editor. Handbook of Human Centric Visualization. New York, NY, USA: Springer, 2012. pp. 285-314.

[28] Freitas CMDS, Pimenta MS, Scapin DL. User centered evaluation of information visualization techniques: Making the HCI InfoVis connection explicit. In: Huang W, editor. Handbook of Human Centric Visualization. New York, NY: Springer, 2012. pp. 315-336.

[29] Fu B, Noy NF, Storey MA. Eye tracking the user experience – an evaluation of ontology visualization techniques. Semant Web 2015; PrePrint, pp. 1-19.

[30] Stojanovic L, Motik B. Ontology evolution within ontology editors. In: Proceedings of the OntoWeb-SIG3 Workshop at the 13th International Conference on Knowledge Engineering and Knowledge Management, 2002. pp. 53-62.

[31] Motta E, Mulholland P, Peroni S, d'Aquin M, Gomez-Perez J M, Mendez V, Zablith F. A novel approach to visualizing and navigating ontologies. In: Proceedings of ISWC 2011, 23–27 October 2011; Bonn, Germany: Springer-Verlag. pp. 470-486.

[32] Katifori A, Halatsis C, Lepouras G, Vassilakis C, Giannopoulou E. Ontology visualization methods—a survey. ACM Computing Surveys 2007; 39: 10.

[33] Lewis JR. Sample sizes for usability studies: additional considerations. Hum Factors 1994; 36: 368-378.

[34] Zapata C, Pow-Sang JA. Sample size in a heuristic evaluation of usability. In: Zapata CM, González G, Manjarrés R, Vargas FA, Arévalo W, editors. Software Engineering: Methods, Modeling, and Teaching. Lima, Peru: Pontificia Universidad Católica del Perú, 2012. pp. 37-36.

[35] Turner CW, Lewis JR, Nielsen J. Determining usability test sample size. International Encyclopedia of Ergonomics and Human Factors. 3, 2006. pp. 3084-3088.

[36] Pretorius MC, Calitz AP, van Greunen D. The added value of eye tracking in the usability evaluation of a network management tool. In: Proceedings of SAICSIT 2005, 20–22 September 2005. White River, South Africa: SAICSIT. pp. 1-10.

[37] Coltekin A, Heil B, Garlandini S, Fabrikant SI. Evaluating the effectiveness of interactive map interface designs: A case study integrating usability metrics with eye-movement analysis. Cartogr Geogr Inf Sc 2009; 36: 5-17.

[38] Conesa J, Storey VC, Sugumaran V. Usability of upper level ontologies: The case of ResearchCyc. Data Knowl Eng 2010; 69: 343-356.

[39] Braun S, Schmidt A, Walter A, Zacharias V. The ontology maturing approach for collaborative and work integrated ontology development. Evaluation, results and future directions. In: Proceedings of ESOE 2007, 12 November 2007; Busan, Republic of Korea. pp. 5-18.

[40] Carvalho RM, Berrios DC, Wolfe SR, Williams J. Ontology development and evolution in the accident investigation domain. In: Proceedings of the IEEE Aerospace Conference, Big Sky, MT, USA. Piscataway, NJ, USA: IEEE, 2005.