



Vocabulary Analysis

Project Client:
Syntax2D
Taubman College of Architecture
& Urban Planning
University of Michigan

Tao Dong
Maureen Hanratty
Adam Torres
Lingyun Xu

Table of Contents

Executive Summary	1
Introduction	2
Overview of Syntax2D	2
Target Population	2
Vocabulary Analysis Goals	3
Methodology	3
Findings & Recommendations	5
Summary of Findings.	5
Detailed Findings	7
Recommendations	10
Discussion	10
Conclusion	10
References	11

Vocabulary Analysis of Syntax2D

Executive Summary

Product and Goals

This report describes the findings of a series of a vocabulary analysis conducted on the software program Syntax2D, an open source software suite for urban and architectural spatial analysis developed at the Taubman College of Architecture & Urban Planning (TCAUP) at the University of Michigan. The goal of this analysis is inspect the vocabulary of the system in terms of four heuristics (order, learning, vocabulary, and timing) and users' expectation. A number of vocabulary issues were identified and solutions were proposed to guide future development of Syntax 2D.

Methods

The research team conducted a general analysis of the vocabulary and an Object/Action analysis. In the general analysis, the researchers walked through the system by reviewing the video obtained from the previous usability testing and took notes from the observation of the system and the user's comments. Four heuristics (order, learning, vocabulary, and timing) were used to identify issues, which were rated later by severity using the Nielson's severity rating framework. The researchers also constructed a grid with objects and actions in the row and column headings in users' language and with commands in the cells in

the system's language. Based on the grid, they compared the vocabulary used by users and the system and identified the discrepancy between the users' expectation and the system's performance.

Findings & Recommendations

Researchers ranked their findings based on the severity of each issue. The major findings and solutions were listed as follows:

1. Finding: The presentation of the "calculate graph measures" option doesn't make clear the consequence of not checking it, and the labeling and placement of the option are problematic.

Solution: Add a hover-over tip or a warning alert to provide additional explanation about the option. In addition, the label should use "grid isovist" instead of "graph" to maintain the consistency of the vocabulary.

2. Finding: The research team found that many users skipped the initialize grid step when attempting to create a grid isovist.

Solution: Consolidate the "Initialize Grid" step with the "Grid Setup/Options" step to reduce error rate and increase efficiency.

3. Finding: The notification about computation time pops up even though the creation of grid isovist hasn't actually been performed.

Solution: Instead of showing "Computation =

0.00”, the system should display error message on the pop-up window to alert the users that the creation of grid isovist has not been performed successfully.

4. Finding: No indication of why measures are grouped together in the drop down menu.

Solution: Adding meaningful headings to the measure groupings will allow users to understand why the measures are grouped as they are.

5. Finding: Many users skipped the “Create Observation Points...” step when attempting to create a path isovist.

Solution: Consolidate this step with the “Path Options...” step. This would also help to show the connection between observation points and the path.

To conclude, the findings and recommendations discussed in this document will help the development team improve the communication between the system and users by speaking users’ language and following specific guidelines (e.g. Grice’s Four Maxims) on making the system more understandable.

Introduction

This report describes the findings of a series of a vocabulary analysis conducted on the software program Syntax2D. Syntax2D is an open source software suite for urban and architectural spatial analysis developed at the Taubman College of Architecture & Urban Planning (TCAUP) at the University of Michigan. The report begins with an overview of Syntax 2D. Next, the target population for the software is described followed by the goals of the vocabulary analysis. An overview of the vocabulary analysis methodology is given in the fourth

section of the report. Detailed findings for the most severe issues with recommendations for improvement are given followed by a summary of the recommendations. In the discussion section the research team reports on an analysis that should be undertaken, by a subject matter specialist, on the system’s use of space syntax terminology.

Overview of Syntax2D

Syntax2D is an open source spatial analysis software program developed by the Taubman College of Architecture and Urban Planning at the University of Michigan. It is a suite of tools for researchers and practitioners to analyze spatial configuration of buildings and urban spaces. Syntax2D features isovist, grid, and axial analysis as well as path analysis and counting features intended to assist with field research. Measures such as connectivity, integration and mean depth can be visualized on a grid and also exported to an Excel spreadsheet as quantitative data. Source files for Syntax2D are computer-aided design (CAD) drawings in the 2000 .DXF format.

Target Population

The target population for the product is university faculty and students in the architecture and urban planning disciplines using space syntax in their research. Space syntax is a relatively small field of study within these disciplines. Research is usually done at the master’s and doctorate level. The University of Michigan has several faculty and Ph.D. and post-doctorate candidates using space syntax in their work. Other centers of space syntax research include the Bartlett School at the University of College London and Georgia Technical University. A space syntax software tool called Depthmap

is currently the mostly widely used software for this type of analysis. The development team hopes to attract current Depthmap users to Syntax2D by providing 1) functionality not available in Depthmap, 2) an ease to use interface, and 3) completely open source code.

Vocabulary Analysis Goals

The goal of the vocabulary analysis was to compare the vocabulary of Syntax2D with the user's vocabulary and expectations. The research team used the information obtained from the vocabulary analysis to discover the areas most in need of improvement and provide a prioritized list of vocabulary issues to address in future versions of Syntax2D.

The goals for vocabulary analysis include:

- ◆ Assess the vocabulary of the system in terms of four heuristics (order, learning, vocabulary, and timing) and identify problematic vocabulary issues in the system
- ◆ Assess the mapping between the system's language and the users' expectation
- ◆ Suggest guidelines for future development of Syntax2D

Methodology

Overview

To achieve the goals, the research team conducted a general analysis of the vocabulary and an Object/Action analysis. The vocabulary studied were around the two core tasks of the system used in the previous usability testing: grid analysis and path analysis.

For the general analysis, the research team

walked through the system by reviewing screen capture footage and audio obtained from a usability test and took observational notes of system menus, dialogue boxes, and the user's comments. Based on these notes, the research team identified a series of vocabulary issues using the four heuristics (order, learning, vocabulary, and timing) and rated the issue by severity using the Nielsen's severity rating framework (Nielsen, 1994).

For the Object/Action analysis, the researchers constructed an Object/Action Grid (see table 1) to get an overview of the system. The problematic object/action pairs were highlighted with different colors according to their severity. With such a grid, one can easily locate the pairs within the contextual system and get a better understanding of the issues and the system.

Four Heuristics

To conduct a general analysis of the vocabulary used by the system, the researchers used four heuristics (fundamentals) for the evaluation found in Mark Newman's PowerPoint: 1) Order--people construct command sentences in their heads and want to speak it (translate it to a system) with as little memory load as possible. This fundamental was used to examine whether the order of the system's actions correspond to the users' command sentences in their heads. 2) Learning--the consistency of the vocabulary within a system and across systems can save a lot of learning time of users. The researchers used this heuristic to examine the consistency of the vocabulary within the system and between this system with other similar systems such as Depthmap. 3) Vocabulary--the team made use of Grice's Maxim of Conversation and Common Ground Theory (Grice, 1975) to examine whether the system is using the words the users know and whether

Table 1. Object/Action Grid

Object	Actions							
	Setup	Initialize	Create Isovist	Create Point	Show Properties	Import	Export	View
Grid	<ul style="list-style-type: none"> Specify total cells Offset Grid Calculate graph measures Integration-n depth 	<ul style="list-style-type: none"> Necessary step before creating grid isovist. 	<ul style="list-style-type: none"> Click 'Create Grid Isovist' in the Grid menu. Click the 'Start' button on the progress popup. 	N/A	<ul style="list-style-type: none"> Click 'Show Grid Properties' Click on measure to be displayed. 	N/A	N/A	N/A
Path	<ul style="list-style-type: none"> Select current path set length specify circle radius and display method for result display 	N/A	<ul style="list-style-type: none"> Click 'Create Path Isovist' 	<ul style="list-style-type: none"> Click on 'Create Observation Points Specify node count OR Specify node spacing 	<ul style="list-style-type: none"> Click 'Show Path Properties' Click on measure to be displayed. 	<ul style="list-style-type: none"> Click on 'Import Path...' Select file to import Specify a layer containing paths. 	N/A	N/A
Layer	N/A	N/A	N/A	N/A	N/A	<ul style="list-style-type: none"> Click on 'Import' in the File menu Click on 'DXF...' Select file to import in the dialog box titled 'Open' Specify a layer names. 	N/A	N/A
Data / Report	N/A	N/A	N/A	N/A	N/A	N/A	<ul style="list-style-type: none"> Click 'Export...' in the Path or Grid menu Select 'Comma Separated Values' Specify a file to export 	<ul style="list-style-type: none"> Click 'View Report...' in the Path or Grid menu View report in a popup titled 'Analysis Summary'

Severity Color Scale	4	3	2	1
----------------------	---	---	---	---

the conversation between the system and the users is informative enough but not redundant.

4) Timing--this heuristic was used to examine whether the response time to the users from the system was appropriate.

Walk Through the System

To fully cover the vocabulary used by the system, the researchers walked through the Syntax2D interface by reviewing the audio and screen capture footage obtained from the previous usability testing. In order to cover the problems of the system as much as possible, the team selected the audio and screen capture footage of the participant who was the most representative of the user base and was the most articulate about problems she encountered during the test. Researchers went through the system using the two tasks constructed for usability testing: grid analysis and path analysis. The research team took notes of possible problems from the observation of the system and the participant's comments.

Object/Action Pairs

Based on observations taken from the usability testing session, the researchers created a series of object/action pairs that were considered problematic. The team used the four heuristics mentioned above to evaluate the object/action pairs and rated them by severity using the Nielsen's (1994) severity rating framework. For each pair, the team proposed specific solutions for the problems identified.

Object/Action Analysis

The researchers constructed an Object/Action Grid to understand the ontology of the system. The grid has objects as its rows and actions as its columns. Within the cells are the problematic object/action pairs that had been identified during the general analysis.

The grid served as a map of the system and gave an overview of the various functionalities of the system (concentrated on the grid and path analysis). By creating such a grid, the team could ensure that all the objects and actions throughout the system were covered.

Moreover, in the grid, the researchers highlighted the problematic object/action pairs with different colors according to the severity of the issues. With the visualization of the context and severity of the problematic pairs, one can get a quick picture of the issues within the system.

Severity Ratings

To prioritize the findings in a consistent way, the researchers utilized Jakob Nielsen's framework of rating severity (Nielsen, 1994):

Rating	Usability Issue
0	I don't agree that this is a usability problem at all.
1	Cosmetic problem only: need not be fixed unless extra time is available on project.
2	Minor usability problem: fixing this should be given low priority.
3	Major usability problem: important to fix, so should be given high priority.
4	Usability catastrophe: imperative to fix this before product can be released.

Findings & Recommendations

Summary of Findings

The research team found a number of issues ranging in severity. A summary of findings is organized in object-action pairs with descriptions

Table 2. Summary of Findings

#	Object-Action Pair	Finding	Severity
1	Setup Options - Grid	The presentation of the “calculate graph measures” option doesn’t make clear the consequence of not checking it, and the labeling and placement of the option are problematic.	4
2	Initialize - Grid	The research team found that many users skipped the initialize grid step when attempting to create a grid isovist.	4
3	Create - Grid Isovist	The computation time window still pops up even if no creation of grid isovist has been performed.	3
4	Show - Grid Isovist Properties	No indication of why measures are grouped together in the drop down menu.	3
5	Create - Observation Points	The research team found that many users skipped the ‘Create Observation Points...’ step when attempting to create a path isovist.	3
6	Create - Grid Isovist	The cancel button is mislabeled as “OK” on the Progress dialog box popped up after clicking “Create Grid Isovist”	2
7	Create - Grid Isovist	The “Start” button is still there after the progress of creating grid isovist has been finished.	2
8	Show - Grid Isovist Properties	Location of Path Count and Point Count in the Grid Properties Menu breaks with system convention.	2
9	Specify - Path Layer Name	Researchers found a dialog box with a title bar that contained directions.	2
10	Show - (Path) Point Properties	Not specifically labeled as "Show Observation Point Properties"	2

of each findings in table 2. The research team utilized the grid creation and path analysis tasks as a method for systematically stepping through the interface. The research team found vocabulary issues related to order, learning, and consistency. The five most severe issues are covered in the Detail Findings.

Detailed Findings

#	Vocabulary Issue	Severity
1	The presentation of the “calculate graph measures” option doesn’t make clear the consequence of not checking it, and the labeling and placement of the option are problematic.	4

Details:

The “Calculate graph measures” option in the “Grid Options” dialog box is a critical option that users have to check off in order to get most of Grid Isovist measures shown. However, during the usability testing, this option was frequently neglected by users. Moreover, when users didn’t get certain measures they cannot relate the error with the option.

The researchers found that the vocabulary ambiguity of this option could contribute to the issue. The option label “Calculate graph measures (Connectivity, etc)” doesn’t provide enough explanation that enables users to make a connection between the option and related measures. This first violates Grice’s Maxim of quantity (Grice, 1975) that “Make your contribution as informative as is required (for the current purposes of the exchange),” because it doesn’t warn of the serious consequence if users don’t check off this option. The labeling

of this option also violates Grice’s Maxim of manner that “Avoid ambiguity” (Grice, 1975). First, it is confusing that the word “graph” isn’t used by any other parts of the program. It makes users difficult to recognize that “grid measures” and “Graph measures” are the same concept. Second, the word “measures” is inconsistent with “properties” which essentially has the same meaning as “measures” but used by other parts of the program, e.g. “Show Grid Properties” (see figure 1).

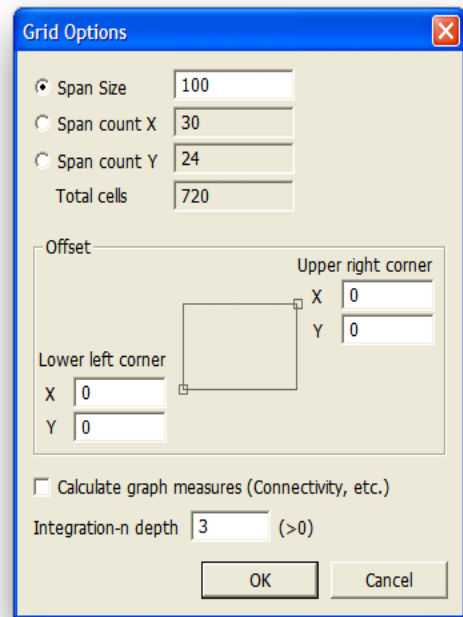


Figure 1. Grid Options dialog box.

In addition, neither the dialog box name “Grid Option” or the menu item title “Grid Setup/Option” imply that there is an option for setting up “Calculating graph measures.” This issue is due to the inconsistency between “graph” and “grid.” Also, users may not expect this option in the “Grid Option” dialog box as its name implies.

Proposed Solution:

First, the development team can add a hover-over tip to provide additional explanation about the option to users. A warning alert may also pop up asking confirmation when users didn't check off this option. Second, the expressions of same concepts should be consolidated in the program. During our usability testing, researchers noticed the participants referred to the "properties" as "measures". It is recommended that the system stick with the term "measure" throughout the program.

As far as the improving the organization of commands, either taking the "Calculate graph measures" out of the grid setup or changing the name of the "Grid Setup/Options" dialog box will work.

#	Vocabulary Issue	Severity
2	The research team found that many users skipped the initialize grid step when attempting to create a grid isovist.	4

Details:

Although the 'Initialize Grid' step is the second selection in the menu, users tended to skip to the 'Create Grid Isovist' menu item (see figure 2). Researchers believe that there is an ordering problem between the first and second step. Furthermore, researchers believe that vocabulary used in 'Initialize Grid' does not convey to the user the importance and necessity of this step.

Proposed Solution:

Nearly all users tested performed the 'Grid Setup/Options...' first, then skipped the 'Initialize Grid' step. We recommend consolidating

the 'Initialize Grid' step with the 'Grid Setup/Options' step for a number of reasons. First, this will reduce the amount of learning that needs to take place before a user can perform a Grid Isovist. Second, the parameters set in the 'Grid Setup/Options' menu are implemented when a user initializes the grid and combining these steps will help strengthen the connection between the parameters set and the

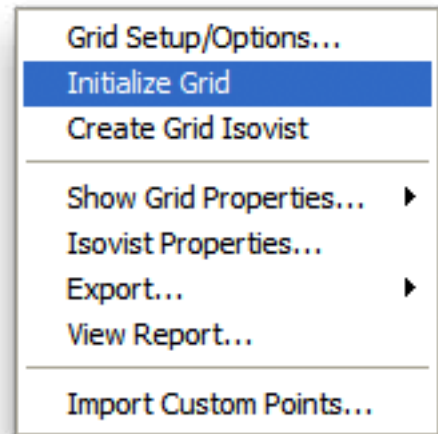


Figure 2. Initialize Grid

#	Vocabulary Issue	Severity
3	The computation time window still pops up even if no creation of grid isovist has been performed.	3

Details:

After the process of creating grid isovist is complete, a window pops up informing the users of the actual time spent. Even if the user does not import any .dxf file into the system or initialize grid, the user will still get the pop-up window showing the computation time. The window

looks the same as the window that appears if the task is executed properly except the computation time is different. Due to the similar look of the windows, the users may not be alert that something has went wrong.

Proposed Solution:

Instead of showing “Computation = 0.00”, the system should display error message on the pop-up window to alert the users that the creation of grid isovist has not been performed successfully.

#	Vocabulary Issue	Severity
4	No indication of why measures are grouped together in the drop down menu.	3

Details:

In the Grid Properties menu, lines are used to separate groups of measures from one another (see figure 3). However, it is not readily apparent why the measures are grouped as they are. In usability testing one participant categorized the second group of measures as syntactic measures. It should be noted that these syntactic measures are not calculated by default: the “Calculate Graph Measures” box must be checked for these measures to be calculated.

Proposed Solution:

Adding meaningful headings to the measure groupings will allow users to understand why the measures are grouped as they are. It will educate users about what kind of information these measures will calculate and display. It may also serve to help users understand that they must select “Calculate Graph Measures” in the “Grid Setup Options” in order the calculate the syntactic measures.

Area (Default)
Perimeter
Drift
Occlusivity
Compactness
Circularity
Connectivity
Mean Depth
Control
Controllability
Relative Asymmetry
Real Relative Asymmetry
Integration
Integration-n
Average Neighbor Control (AN)
Global E (GE)
Global Relative E (GRE)
Global Connectivity E (GCE)
Depth from current cell
Mean Radial
Max Radial
Min Radial
Complexity (Radial Variance)
Elongation
Autocorrelation
Point Count
Path Count
User defined...

Figure 3. Measures available in Grid Properties

#	Vocabulary Issue	Severity
5	The research team found that many users skipped the 'Create Observation Points...' step when attempting to create a path isovist.	3

Details:

Researchers believe that creating observations points is an option that needs to be specified during the path options step.

Proposed Solution:

When a user creates observations points, they are directly affecting the path that they had previously selected, which leads us to believe that it should be consolidated with the 'Path Options...' step. This would also help to show the connection between observation points and the path.

Recommendations

The findings detailed above suggested improving Syntax 2D in three aspects: labeling, ordering and grouping. To solve the labeling issues in Syntax 2D, the development team need to fix false labeling (e.g. change the button "OK to "Cancel" on the progress bar before creating a Grid Isovist), supply necessary information on label texts (e.g. add sufficient explanation about the effect of the "Calculate Graph Measures" option), and make labeling consistent (e.g. replace "properties" with "measures"). The development team should also follow users' mental models to adjust the order of commands both in the menu and the toolbar in order to meet users' expectation. Last but not least, the grouping of commands could be used to consolidate some redundant actions (e.g.

"Initialize grid") so as to reduce the possibility of skipping a step as well as improve efficiency. In a word, the development team needs to improve the communication between the system and users by speaking users' language and following specific guidelines (e.g. Grice's Four Maxims) on making the system more understandable.

Discussion

The research team recommends that the Syntax2D development team review the space syntax specific terminology used in the system with a space syntax subject matter expert. During usability testing one participant reported that she did not know the meaning of many of the measures that are available in Syntax2D. She also stated that, as space syntax is an emerging field, there is dispute about the terminology used for specific calculations. The Syntax2D team should regularly inventory the terminology used in the system to ensure that it follows conventions established in the research literature and community.

Conclusion

Overall Syntax2D meets user's expectations by organizing the system, including toolbars and file menus, on a set of well understood objects: the grid, the path, the layer, and the data. However the research team found problems with the ordering of actions, the use of ambiguous language, and the use of uninformative language. There were also situations where language, hitherto missing, could be used to disambiguate groupings of terms. The research team recommends further analysis be conducted on the space syntax terminology used throughout the system.

References

Nielsen, J. (1994). Heuristic evaluation. In J. Nielsen & R. L. Mack (Eds.), *Usability inspection methods* (pp. 25-62). New York: John Wiley & Sons.

Grice, H.P., (1975). *Logic and Conversation*. in Cole, P. & Morgan, J. (Eds.) *Syntax and Semantics 3: Pragmatics*. New York: Academic Press.