Usability Studies on a Visualisation for Parallel Display and Control of Alternative Scenarios

Aran Lunzer & Kasper Hornbæk
Natural Sciences ICT Competence Centre
University of Copenhagen, DK-2100 Copenhagen, Denmark
{alunzer, khornbaek}@nik.ku.dk

ABSTRACT
Many applications require comparison between alternative scenarios; most support it poorly. A subjunctive interface supports comparison through its facilities for parallel setup, viewing and control of scenarios. To evaluate the usability and benefits of these facilities, we ran experiments in which subjects used both a simple and a subjunctive interface to make comparisons in a census data set. In the first experiment, subjects reported higher satisfaction and lower workload with the subjunctive interface, and relied less on interim marks on paper. Subjects also used fewer interface actions. However, we found no reduction in task completion time, mainly because some subjects encountered problems in using the facilities for setting up and controlling scenarios. Based on a detailed analysis of subjects' actions we redesigned the subjunctive interface to alleviate frequent problems, such as accidentally adjusting only one scenario when the intention was to adjust them all. At the end of a second, five-session experiment, users of this redesigned interface completed tasks 27% more quickly than with the simple interface.

Categories and Subject Descriptors
H.5.2 [Information Interfaces and Presentation]: User Interfaces – Input devices and strategies, Interaction styles, Evaluation/methodology

General Terms
Human Factors, Experimentation

Keywords
Subjunctive interfaces, information visualisation, usability study, iterative design and evaluation

INTRODUCTION
Few things in life can be evaluated in isolation. Comparison of alternative scenarios is indispensable, for example in information exploration, when comparing travel plans that use different airlines; in design, such as when investigating the influence of image placement on the layout of a web page; and in simulation, when testing how alternative population growth scenarios would affect a country's economy. In particular for complex tasks, which require non-trivial problem solving and have no fixed route to their solution, there is a need for what-if exploration of scenarios of interest, and for interfaces supporting comparison of those scenarios [11,15].

Many applications do support some degree of comparison between scenarios: information visualisation interfaces [3] may be used for building visualisations that highlight differences, and in direct manipulation interfaces [14] the user can explore alternatives with the help of immediate feedback and reversible actions. However, Terry and Mynatt [17] point out that most applications are still anchored to a 'single-state document model', making parallel and flexible exploration of alternative scenarios difficult. They suggest that new, generally applicable interface mechanisms are needed to give users better support for experimentation, variation and evaluation. Terry and Mynatt's work on side views [18] is one effort towards such mechanisms, where users can create persistent previews of the effects of commands, or ranges of parameters. Another effort is subjunctive interfaces [9,10], which help users to set up, view and control alternative scenarios based on different input-parameter values. To our knowledge, however, neither of these approaches has been evaluated empirically.

This paper describes two experiments that provide initial data on the usability of subjunctive interfaces. The experiments are based on the browsing of census data, as one activity that would naturally call for comparisons. We analyse in detail users' interactions with the subjunctive interface provided in the first experiment, and identify some recurrent problems. A revised design informed by this analysis is used in a second experiment, which also investigates how longer-term use affects users' performance with the subjunctive interface.

Next, we give an example of a subjunctive interface and discuss related work. Then we present the two experiments, and conclude by briefly discussing the results.

A SUBJUNCTIVE INTERFACE
We introduce the principles of subjunctive interfaces through a description of the two census browsers that we compared in our first experiment. Figure 1 shows a browser based on the 'simultaneous menus' interface used in [8], for browsing data on commercial activity in the state of Maryland. The data set contains 828 records, holding the statistics for nine industry areas in each of twenty-three counties over four successive years. Each record specifies the number of employees, the number of establishments, and the total annual payroll. The user specifies a record by making selections in three menus (1.1 to 1.3); the statistics appear as results in area 1.4.

AVI '04, May 25-28, 2004, Gallipoli (LE), Italy

Figure 1. The simple interface for browsing census data. It is based on the simultaneous-menus design that was shown in [8] to be more effective than sequentially presented menus. For a selected county (1.1), industry (1.2), and year (1.3), the results area (1.4) shows the number of employees, total annual payroll, and number of establishments.

Figure 2 shows a subjunctive interface for browsing the same data set. Its facilities exemplify the three principles of subjunctive interfaces, as follows:

First, the user should be able to set up multiple, independent scenarios that may be mutually incompatible. When browsing census data, a scenario comprises a set of selections (county, industry and year) and the display of the corresponding results. Say a user wants to compare the results from different years. With the browser in Figure 1 (referred to in our experiments as the 'simple interface', because it supports just one scenario), the user must click each year in turn and read off that year's results. With the subjunctive interface, the years can be set up in parallel scenarios. Panels b and c in Figure 2 show how a user sets up new scenarios as copies of existing ones.

Second, the scenarios should be viewable simultaneously, in a way that helps the user to compare them and to see which values belong to which scenario. With the simple interface, comparing census results requires the user to remember result values. In the subjunctive interface, the results appear side by side; Figure 2a shows four scenarios (for two counties in each of two years). Correspondence between the menu selections and the results for each scenario is shown by position and colour cues in the result displays and in the markers next to menu items.

Third, the user should be able to control scenarios in parallel, so that an adjustment to an input parameter can be applied to more than one scenario at a time. In census browsing, the input parameters are the menu selections. With the simple interface, a change to a menu selection affects the only scenario; in the subjunctive interface, the user may affect multiple scenarios with each change. The user can select which scenarios are 'active', meaning that mouse operations will affect them. In Figure 2a the bottom two scenarios (those for 1994) are active; if the user wishes to change the year of these scenarios to 1996, this requires just one click on 1996. Additionally, by holding down the Alt key the user can force all scenarios to be changed at once; for example, changing them all from Construction to Manufacturing with a single Alt-click on Manufacturing.

This is just one example of a design implementing the three principles of a subjunctive interface. Other approaches are possible, such as overlaying the scenarios' displays or using different visualisations of the results. For descriptions of such design choices see [9,10].

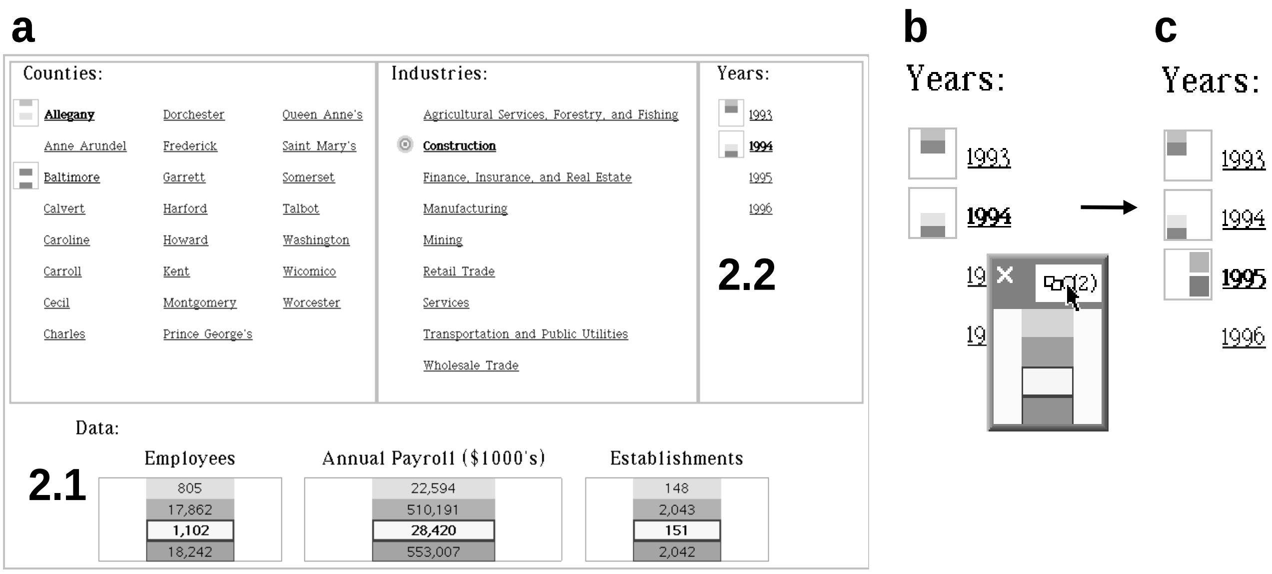

Figure 2. The subjunctive interface, with four scenarios holding the Construction statistics for both Allegany and Baltimore, in 1993 and 1994. Correspondence between menu selections and result values is indicated with position and colour cues in the result displays (2.1) and the markers next to menu items (e.g., 2.2); for example, the values 805, 22594 and 148 at the top of the result displays are for Allegany in 1993. The bottom two scenarios are currently 'active', i.e., affected by mouse operations. Panel b shows the user copying these two scenarios, by clicking and holding the mouse on 1995 and selecting the copy icon at top right in the resulting pop-up; panel c shows how the Years menu will appear with the new scenarios for 1995.

RELATED WORK

Our general concern can be seen as helping users to place alternatives into context. This concern arises in many fields within HCI, e.g. in information visualisation [3] and direct manipulation interfaces [14]. For viewing and comparing alternatives within tabular data, for example, interfaces such as Polaris [16] provide rich facilities for constructing and reconfiguring the table. The Table Lens [12] lets a user visualise chosen rows relative to each other and to the full range of values in each column. These tools, however, are limited to data suitable for row-and-column display, while a subjunctive interface can handle alternatives for any input or display region in an interface.

Tools such as the Attribute Explorer [20] and Spotfire [1] project numerical or ordinal data onto a two-dimensional graphical layout, and provide interactive controls that allow a user to highlight data elements or ranges. Comparison is supported by the user's ability to switch the display rapidly and reversibly among different settings for the highlighting. Such dynamic switching is good for drawing attention to subtle distinctions, especially along some continuous range, but in other cases comparison may be better supported by simultaneous, side-by-side presentation of a set of key cases. Tufte [19] supports the use of replicated 'multiples' that show different values using similar format, saying that 'Multiples directly depict comparisons, the essence of statistical thinking' (p. 105). Roberts [13] likewise recommends view multiplicity in computer interfaces as a way to encourage users to try out alternatives. Using multiples allows a subjunctive interface to lay out scenarios for comparison, even when the elements to be compared, such as text blocks or complex graphical views, have no default, readable way of being arranged within a single two-dimensional view.

Design considerations for subjunctive interfaces are related to those for multiple view visualisations [2]. The Information Visualization Spreadsheet [4] brings the principles of numerical spreadsheets to the building of visualisations; like in a normal spreadsheet, setting up parallel views to show independent derivations is just a matter of copying the cells that make up the derivation chain. Our subjunctive interface for census browsing can be seen as a specialisation of such parallelism, where instead of providing general-purpose, copiable cells we augment specific display regions (the menus and the displays) to allow one or more scenarios' values to be shown.

The menu markers and result displays in our subjunctive interface use correlated layout and colouring to reveal which scenarios have which values. These cues are similar to the use of brushing and linking within other visualisation systems (e.g., Visage [6]), where a data object coloured by the user in one view carries that same colour wherever else it appears.

EXPERIMENT #1

Our overall goal was to evaluate the usability of subjunctive interface facilities in tasks requiring comparison between scenarios. For this experiment we therefore asked users to perform a variety of pre-defined comparison tasks, using each of the two interfaces described above. We formed the following hypotheses:

H1: With the subjunctive interface, users will rely less on writing interim marks and notes, e.g., to keep track of the results of interest or of progress in performing a complex task. This hypothesis reflects the goal of information visualisation to amplify cognition through support of external cognition [3].

H2: Users will complete tasks with fewer mouse clicks when using the subjunctive interface [10]. As a consequence, we also expect users to be faster with the subjunctive interface.

H3: Users will have higher satisfaction with and prefer the subjunctive interface. In particular, we expect users to appreciate the direct on-screen comparisons possible with the subjunctive interface.

H4: Users will report lower mental workload using the subjunctive interface, because they do not have to remember the values that are to be compared.

Subjects

Twenty paid subjects participated in the experiment: sixteen men and four women. Subjects were recruited among students and faculty, and had a mean age of 32 years.

Tasks

The tasks reported in [8] were all simple two-case comparisons, that on average could be solved in about 30 seconds. For this experiment we defined more complex retrieval and comparison tasks, of the following three types:

Intra-set comparison: These tasks require pair-wise comparisons between many combinations of the records in some small set. For example, one task asks: 'Considering Wholesale Trade in [five named counties] in 1993, find how the counties are ordered in terms of number of Employees. In order from fewest Employees to most, what are the Payroll values for these counties?'

Iterative examination: These tasks call for examination of records that lie in a repeating pattern. For example, 'In 1996, for which of the industries do [three named counties] all have 1000 or more Employees?'

Iterative comparison: These tasks are similar to iterative examination, but call for comparison between the records rather than merely checking whether each record satisfies some criterion. For example, 'In which counties does the Payroll for Wholesale Trade fall in every year from 1993 to 1996?'

We expected that, for each task type, appropriate use of the subjunctive interface would provide some benefit over using the simple interface. For intra-set comparisons the benefit is merely in being able to keep values on view rather than having to remember them or write them down. For iterative examinations and comparisons, the iteration can be performed more efficiently if the user first sets up scenarios that express the repeating pattern required by the task.

Design and measures

We used a within-subjects design, where each subject first solved a set of nine tasks (three of each of the above types) using one interface, then solved another nine with the other interface. The order in which subjects used the interfaces, and the order of the two task sets, were systematically varied; each subject was assigned randomly to one of four groups determining these orders.

The independent measures are the two interfaces (simple vs. subjunctive), and the three task types. The dependent measures are the following:

- accuracy in answering tasks, measured as the percentage of tasks correctly answered;
- task completion times, excluding time to read the task;

- subjective satisfaction, measured by five questions from the Questionnaire for User Satisfaction, QUIS [5];
- experienced mental workload, measured by NASA's task load index (TLX) questionnaire [7];
- preference, measured by asking at the end of the experiment which interface the subject preferred.

We logged with time stamps all interface actions (mouse clicks and keystrokes) performed by the subjects.

Procedure

Upon arriving, subjects filled in a questionnaire with background information. Next, they received a standardised oral explanation of how to use the interfaces, and completed four simple training tasks. Any questions or misunderstandings brought up at this stage were resolved. In all, training took an average of 29 minutes.

For each interface, subjects received one task at a time on a piece of paper on which they could also write the answer. Subjects could request clarification of what information the task demanded, but not of how to find or record that information. They were given a maximum of eight minutes for each task; if the task was not completed after this time, the subject was asked to move on to the next. Subjects could also decide to abandon a task. Out of the total 360 tasks, two were abandoned and one timed out; these three cases have been excluded from the statistical analysis.

Having completed all the tasks for an interface, subjects were given five questions from the QUIS [5] and an opportunity to comment on that interface. Next, they completed NASA's TLX questionnaire [7] as a measure of mental workload. Between using the two interfaces, subjects were given a five-minute break.

After using both of the interfaces, subjects wrote down which interface they preferred and why.

RESULTS OF EXPERIMENT #1

Table 1 and figures 3 and 4 summarise the outcome of the experiment. Below we analyse each dependent variable in turn using analysis of variance with repeated measures.

	Simple interface (N=180)	Subjunctive interface (N=177)
Percent correct answers	89% (31)	86% (35)
Number of marks written on paper*	2.62 (4.74)	.83 (2.84)
Number of interface actions*	34.1 (21.3)	21.9 (16.0)
Task completion time (s)	135.0 (64.2)	138.4 (72.2)
	(N=20)	(N=20)
Preference*	2	18

Table 1. Usability and usage differences between interfaces. Parentheses give the standard deviation. Asterisks indicate significant differences with p<.001.

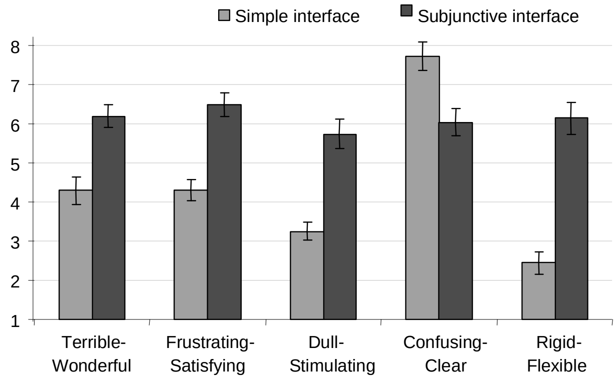

Figure 3. Subjective satisfaction with the interfaces. On all questions there exists a significant difference between interfaces (high scores associated with positive words).

Accuracy

We find no overall difference in accuracy between the interfaces, $F(1,19)=1.14$, $p>.3$. With the subjunctive interface, subjects correctly answered 86% of tasks; with the simple interface, 89%.

Number of marks written

In the course of answering a task, subjects using the simple interface made more marks on the paper than subjects using the subjunctive interface, $F(1,19)=17.32$, $p<.001$. Counting as a single mark anything with an isolated meaning – whether a simple tick or a written data value – subjects using the simple interface made 2.62 marks per task, compared with 0.83 marks for the subjunctive interface. Furthermore, marks were only used in approximately 10% of tasks solved with the subjunctive interface, as against 32% with the simple interface. Note that when explaining the experimental procedure to subjects we did not mention the option of making extra marks on the paper.

Efficiency - Number of interface actions

Overall, the data confirmed our hypothesis that the subjunctive interface would require fewer actions for task completion, $F(1,19)=81.71$, $p<.001$. On average, subjects with the subjunctive interface used 22 actions to complete a task; with the simple interface, subjects used 34 actions.

Efficiency - Task completion time

We find no significant difference in task completion time, $F(1,19)=0.3$, $p>.8$, so hypothesis H2 is not confirmed. On analysing individual tasks we find differences for three tasks of the most complex type (iterative comparisons). Of these, two are solved 48% and 90% more quickly with the subjunctive interface, while conversely the third is solved 56% more slowly with the subjunctive interface.

Notably, the lower number of interface actions with the subjunctive interface does not lead to lower task completion times.

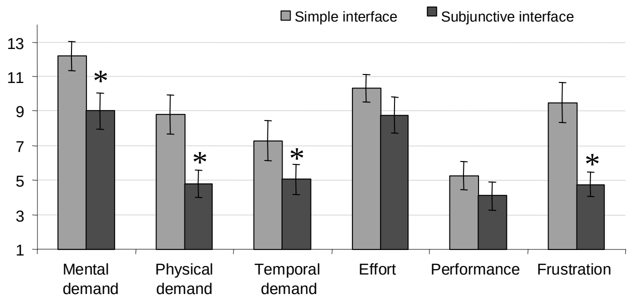

Figure 4. Workload with the interfaces, measured by TLX [7]. A significant difference that favours the subjunctive interface was found for the four questions that are marked with asterisks (high scores associated with high workload).

Subjective satisfaction

Eighteen subjects preferred the subjunctive interface and two preferred the simple interface – a highly significant and large difference: $\chi^2(1, N=20) = 12.8, p<.001$.

Figure 3 summarises the responses to the subjective satisfaction questions. On four questions the subjunctive interface is rated significantly higher than the simple interface; on one question (how confusing or clear the interface is) the simple interface is rated higher. All these differences are significant using paired t-tests with Bonferroni adjustment (terrible-wonderful: t=-5.25, p<.01; frustrating-satisfying, t=-5.40, p<.01; dull-stimulating, t=-5.42, p<.01; confusing-clear, t=3.61, p<.05; rigid-flexible, t=-7.96, p<.01).

Experienced mental workload

On NASA's TLX, subjects assessed the subjunctive interface as requiring less workload on all dimensions (see Figure 4). Overall, there is a significant difference between interfaces, suggesting that the subjunctive interface requires less mental workload, $F(1,19)=12.14$, p<.01. Individual tests show significant differences for the item on mental demand (t=2.51, p<.01), physical demand (t=3.45, p<.01), temporal demand (t=2.27, p<.05), and frustration (t=3.15, p<.01).

The comments given by subjects on post-experiment open-ended questionnaires confirm the above results. Ten subjects commented that the subjunctive interface supported easy comparison, required less remembering of values and helped to avoid writing values down. One subject wrote: 'it was clearly an advantage to be able to see more [values] at once, so that you did not have to do maths in the head or count using your fingers'. Similarly, four subjects commented that the main drawback of the simple interface was the need to remember. Another frequent comment, made by seven subjects, was that they had too little time to learn the subjunctive interface. For example, one subject said 'I felt that I need more time to be familiar with the [subjunctive] interface to be able to work faster and have a higher satisfaction'.

DISCUSSION OF EXPERIMENT #1

Three of our hypotheses were confirmed. Subjects using the simple interface relied more on pen and paper to remember values and to organise their search, confirming H1. Subjects preferred the subjunctive interface and reported markedly lower mental workload with that interface, confirming H3 and H4 respectively.

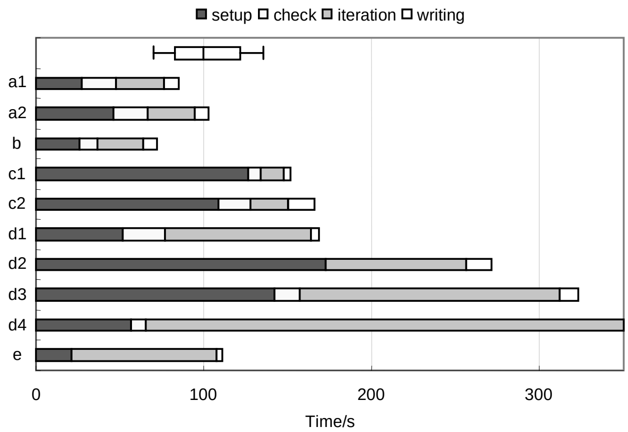

Figure 5. Timing for ten subjects' solutions of an iterative-examination task using the subjunctive interface. Using the interface logs we have divided progress into four commonly seen phases: *setup* to create the scenarios for iteration; *check* to confirm by inspection that these scenarios reflect the task instructions; *iteration* when iterating over the required range; and *writing* between the last interface action and asking for the next task. The y axis associates subjects with five kinds of strategy (a to e) that were pursued. The box plot at the top shows the times for solving this task with the simple interface.

However, although the average number of interface actions per task was significantly lower for the subjunctive interface, the task completion times relevant to hypothesis H2 were less conclusive.

It seems that the overall reason for the subjunctive interface's poor timings is that many subjects encountered difficulties in using it. From notes taken during the experiment, and later analysis of the detailed interface logs, we define difficulties of two kinds: strategy formation problems, and strategy execution problems.

Strategy formation problems

To solve a task efficiently, the user must first decide on a good strategy for using the available interface. When using the subjunctive interface, many subjects had difficulty figuring out a good way to use its relatively complex facilities. In particular, optimal solution of the iterative tasks requires understanding the practice of setting up scenarios that differ in one parameter, then iterating all of them through the values on a second parameter.

We illustrate some of these problems by showing the solution efforts on one task that elicited a wide range of successful and unsuccessful responses. The task was: 'Anne Arundel, Carroll, Harford and Howard are four counties in the region called Central Maryland. In 1996, which of these counties had over 10,000 Employees in three or more of the industries?' Figure 5 shows the timing for the ten subjects who tackled this task using the subjunctive interface.

For this task, an effective strategy is to create a nine-scenario display showing the records for all the industries in one county. The nine Employees values can rapidly be checked by eye to see whether three or more meet the required criterion, and the entire display can be iterated through the other counties. Just two of the ten subjects (a1 and a2 in Figure 5) devised this strategy from the outset and successfully pursued it to find the correct answer.

A less convenient strategy is to create scenarios for each of the counties, then iterate that pattern through the industries. The subject must keep a mental or written record of whether three qualifying Employees values have been seen yet for each county. Unfortunately, seven of the subjects began their approach to this task by creating this four-county setup. One subject (b) successfully carried the strategy through just using his memory. Two others (c1 and c2) set up the four county scenarios then paused, reset the interface and restarted using the nine-industry approach instead; this is seen in their long setup times but short iterations. Four subjects (d1 to d4) did not make this strategy switch, but apparently tried to adapt the four-county setup to keep on view the records that turned out to meet the task criteria. This was not a effective strategy. Of these four subjects, d4 eventually abandoned the task; the others ended up iterating multiple times, as they might have done with the simple interface. One subject (e) pursued an unusual task-specific strategy, creating scenarios to hold what had turned out to be the largest industries in the first examined county – presumably in the hope that these would also give a quick positive answer for other counties.

Strategy execution problems

In many cases, subjects who had apparently thought up an effective strategy then ran into difficulties in executing it. Problems of this type might be regarded as being specific to the census browser, but where they relate to its scenario-handling facilities they reveal issues relevant to any subjunctive interface.

The most common problem was when a subject intended and expected a single mouse-click to affect all scenarios (for example, to change them all from one industry to another), but the click only affected one scenario. As described earlier, Alt-click affects all scenarios, but otherwise a click only affects the scenarios currently selected as 'active'. Subjects often failed to notice which scenarios were active, and failed to use the actions they had been taught for changing that selection.

Exacerbating this problem, sometimes subjects simply did not understand what the interface was showing. One source of confusion was the way of displaying 'empty scenarios' – i.e., scenarios that lack a selection in one or more of the menus, and therefore have no results. For example, a subject might start a task by creating four scenarios that have different years, but no county or industry selection as yet. Then clicking on the county and industry menus without using the Alt key would set values in a single active scenario, leaving the other three scenarios empty. Subjects were confused to see result views that each contained one number and three blanks.

Confusion also arose because of a design decision that a result display should show just a single value, full size, if all scenarios contain that same value. In theory this can save the user effort in recognising when a value is the same in all scenarios, but in practice subjects were sometimes confused if they set up multiple scenarios that just happened to be identical. Why wasn't the interface showing multi-valued result displays?

Finally, a number of subjects misunderstood that once a selection is made it cannot be cancelled, but only moved to a different value. Subjects who made an unintentional selection sometimes tried to remove it by pressing the Delete key, which has a more destructive effect than was probably intended: deleting one or more entire scenarios. This is indicative of the general problem that the subjects were not sufficiently proficient with the interface to correct mistakes or slips when they arose.

EXPERIMENT #2

Subjects in Experiment #1 commented that, in the brief time they used the subjunctive interface, they could not become competent with it. The strategy formation and strategy execution problems that we observed may therefore not be persistent. To investigate this, we conducted a second experiment in which subjects used a redesigned version of the interface over five sessions. Our main hypothesis was that, over sessions, the subjunctive interface would become significantly faster than the simple interface.

Interface changes

The subjunctive interface for Experiment #2 incorporates several changes to address the execution problems described above. The main changes are as follows:

Which scenarios are affected by a mouse click

The use of active scenarios and Alt-clicks has been replaced by a mechanism based on highlighting just one of the item markers within each menu. Two rules now determine which scenarios are affected by a click on a menu item: (1) When there are no item markers in the menu (because none of the items has been selected yet), a click affects all scenarios; (2) Once the menu has some selections, a click affects just the scenarios corresponding to the currently highlighted item marker. The user can highlight a different marker by clicking on it.

How to copy scenarios

As part of the expanded role of the item markers, scenario copying is now performed by dragging a marker from one menu item and dropping it on another while keeping the Ctrl key pressed. This copies all the scenarios in that marker, and selects the drop-target item's value in the new scenarios. For consistency, dragging without pressing Ctrl also has a meaning: as the mouse passes over an item, its value is selected in the scenarios of the dragged marker.

Display of empty scenario results

Instead of blanks, the result displays for an empty scenario show textual prompts (e.g., '**Industry? Year?**') to help the user understand what selections are still needed. This change was also applied to the simple interface.

Display of identical scenario results

Results from multiple scenarios are laid out in full, even if their values happen to be identical.

Subjects

Seven subjects who participated in the first experiment and who were willing to participate again were paid to take part.

Design and procedure

The experiment consisted of five sessions, each separated by at least a day. As in Experiment #1, subjects used both interfaces in each session. In Session 1, subjects were trained and completed nine tasks with each interface. Sessions 2 to 4 were completed at places of the subjects' choosing. They received a CD with the software together with stapled task sets containing nine tasks for each interface. In Session 5, subjects came back to the lab and completed the tasks and the satisfaction questionnaire used in Experiment #1.

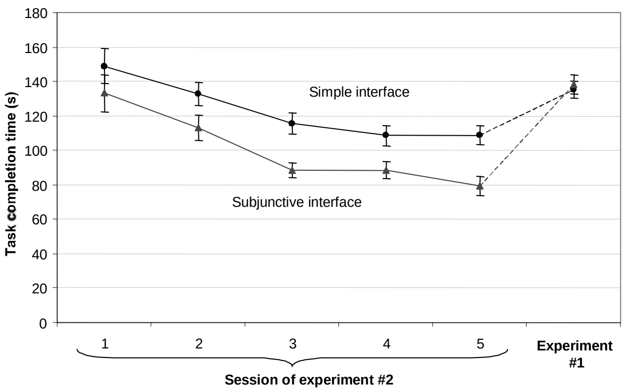

Figure 6. Task completion times in experiments 2 and 1. Error bars indicate standard error of the mean. Each session consists of N=126 observations.

Tasks

The tasks used for Sessions 1-4 in Experiment #2 were similar to those used in Experiment #1. To avoid subjects simply remembering the answers to tasks, we constructed four sets of tasks with the same structure as in Experiment #1 but concerning different counties, industries, years, and kinds of comparison (e.g., finding the year with the highest payroll instead of the lowest).

Results of experiment #2

Subjects' accuracy in answering the tasks correctly is in general high, and not different between interfaces. In addition, all subjects now show preference for the subjunctive interface. To address our main hypothesis, we therefore focus on task completion time.

Figure 6 shows the average task completion time over sessions. Overall, analysis of variance using repeated measures shows that the subjunctive interface is faster than the simple interface, $F(1,6)=8.27$, $p<.05$. Planned comparisons show this to hold for all sessions except Session 1. In Session 4, subjects are on average 18% faster with the subjunctive interface (Subjunctive interface: M=89s, SD=39; simple interface: M=109s, SD=46).

Comparing Session 5 to Experiment #1, which used the same tasks, we find a significant improvement in task completion times for both interfaces. However, in Session 5, the subjunctive interface is approximately 27% faster than the simple interface, $F(1,6)=208.87$, $p<.001$. In Experiment #1, we found differences at the task level favouring either the subjunctive or the simple interface; in Session 5 of Experiment #2, we find six tasks that are solved significantly faster with the subjunctive interface, and no tasks that subjects completed significantly faster with the simple interface. For one task, subjects on average took more than 2.3 times as long with the simple interface.

Discussion of experiment #2

The results of Experiment #2 strongly confirm our hypothesis: over sessions, subjects became faster with the subjunctive compared to the simple interface. It was also evident that strategy formation problems diminished as subjects became more experienced at controlling the subjunctive interface.

CONCLUDING REMARKS

Subjunctive interfaces support parallel setup, viewing and control of alternative scenarios, and can therefore be expected to benefit tasks requiring inter-scenario comparison. Through two usability studies we have shown that, with careful design, a subjunctive interface can indeed give large, statistically significant performance benefits for comparison tasks. The studies show that subjects preferred the subjunctive interface over the original, simple interface, and rated it as being more satisfying to use. With the simple interface subjects also depended more on writing down or remembering data, as suggested by more interim marks made on paper and by the higher mental workload reported. After several hours' experience with both interfaces, subjects were completing tasks 27% more quickly with the subjunctive interface than with the simple interface.

As comparison of alternative scenarios is indispensable in many applications, we believe that software designers in general should consider extending their applications based on subjunctive-interface principles. Often such extension would require just an upgrade of the existing interface, not a full replacement.

One goal for further empirical evaluation is to see whether complex, possibly creative tasks lasting hours or days can also benefit from the provision of a subjunctive interface; another is to study more closely users' decisions to create and use multiple scenarios. Further development of subjunctive-interface mechanisms should address how they can scale to support activities of greater complexity – larger numbers of input parameters and choices on those parameters, and larger numbers of scenarios to be displayed simultaneously.

ACKNOWLEDGEMENTS

We thank Harry Hochheiser for lending us the census data, and Erik Frøkjær for comments on an early draft of this paper. The interfaces used in this experiment were built using the Morphic library within Squeak Smalltalk.

REFERENCES

[1] Ahlberg, C. and Shneiderman, B. Visual information seeking: tight coupling of dynamic query filters with starfield displays. In *Proceedings of the SIGCHI conference on human factors in computing systems (CHI '94)*, ACM Press, New York, NY, 1994, 313-317.

[2] Baldonado, M. Q. W., Woodruff, A., and Kuchinsky, A. Guidelines for using multiple views in information visualization. In *Proceedings of the international conference on Advanced Visual Interfaces (AVI 2000)*, ACM Press, New York, NY, 2000, 110-119.

[3] Card, S. K., Mackinlay, J. D., and Shneiderman, B. *Readings in Information Visualization.* Morgan Kaufmann, San Francisco, CA, 1999.

[4] Chi, E. H., Riedl, J., Barry, P., and Konstan, J. Principles for information visualization spreadsheets. *IEEE Computer Graphics and Applications, 18,* 4 (Jul./Aug. 1998), 30-38.

[5] Chin, J. P., Diehl, V. A., and Norman, K. L. Development of an instrument for measuring user satisfaction of the human-computer interface. In *Proceedings of the SIGCHI conference on human factors in computing systems (CHI '88),* ACM Press, New York, NY, 1988, 213-218.

[6] Derthick, M. and Roth, S. F. Data exploration across temporal contexts. In *Proceedings of the international conference on Intelligent User Interfaces (IUI 2000)*, ACM Press, New York, NY, 2000, 60-67.

[7] Hart S. G. and Staveland, L. E., Development of NASA-TLX (Task Load Index): results of empirical and theoretical research. In Hancock, P.A. and Meshkati, N. (eds.), *Human Mental Workload*. Elsevier, 1988, 139-183.

[8] Hochheiser, H. and Shneiderman, B. Performance benefits of simultaneous over sequential menus as task complexity increases. *International Journal of Human-Computer Interaction*, *12*, 2 (2000), 173-192.

[9] Lunzer, A. Towards the subjunctive interface: general support for parameter exploration by overlaying alternative application states. In *Proceedings of the IFIP TC. 13 international conference on human-computer interaction (INTERACT '99)*, IOS Press, Amsterdam, The Netherlands, 1999, 474-482.

[10] Lunzer, A. and Hornbæk, K. Side-by-side display and control of multiple scenarios: subjunctive interfaces for exploring multi-attribute data. In *Proceedings of the Australian conference on computer-human interaction (OZCHI 2003)*, Brisbane, Australia, 2003, 202-210.

[11] Mirel, B. *Interaction Design for Complex Problem Solving. Developing Useful and Usable Software*. Elsevier/Morgan Kaufmann, 2003.

[12] Rao, R. and Card, S. The table lens: merging graphical and symbolic representations in an interactive focus+context visualization for tabular information. In *Proceedings of the SIGCHI conference on human factors in computing systems (CHI '94)*, ACM Press, New York, NY, 1994, 318-322.

[13] Roberts, J. C. Multiple-view and multiform visualization. In *Proceedings of the SPIE conference on visual data exploration and analysis VII (IS&T/SPIE 2000), Proceedings of SPIE*, 3960 (Jan. 2000), 176-185.

[14] Shneiderman, B. Direct manipulation: a step beyond programming languages. *IEEE Computer*, *16*, 8 (Aug. 1983), 57-68.

[15] Shneiderman, B. *Leonardo's Laptop: Human Needs and the New Computing Technologies***.** MIT Press, 2002.

[16] Stolte, C., Tang, D., and Hanrahan, P. Polaris: a system for query, analysis and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics*, *8*, 1 (Jan. 2002), 52-65.

[17] Terry, M. and Mynatt, E. Recognizing creative needs in user interface design. In *Proceedings of the fourth conference on Creativity and Cognition (C&C 2002)*, ACM Press, New York, NY, 2002, 38-44.

[18] Terry, M. and Mynatt, E. Side views: persistent, on-demand previews for open-ended tasks. In *Proceedings of the ACM symposium on User Interface Software and Technology (UIST 2002),CHI Letters*, *4*, 2 (2002), 71-80.

[19] Tufte, E. *Visual Explanations*. Graphics Press, Cheshire, CT, 1997.

[20] Tweedie, L., Spence, B., Dawkes, H., and Su, H. Externalising abstract mathematical models. In *Proceedings of the SIGCHI conference on human factors in computing systems (CHI '96)*, ACM Press, New York, NY, 1996, 406-412.