Automated Assistance in Evaluating the Layout of On-screen Presentations

Karin Harbusch¹, Denise Dünnebier¹, Denis Krusko¹

¹Computer Science Department, University of Koblenz-Landau, Germany
{harbusch|duennebier|kruskod}@uni-koblenz.de

Keywords: Human-Computer Interaction, Interface Design, Personalized Feedback, Presentation Design, Presentation Layout, Evaluation Assistant System

Abstract: Oral presentations can profit decisively from high-quality layout of the accompanying on-screen presentation. Many oral talks fail to reach their audience due to overloaded slides, drawings with insufficient contrast, and other layout issues. In the area of web design, assistant systems are available nowadays which automatically check layout and style of web pages. In this paper, we introduce a tool whose application can help non-experts as well as presentation professionals to automatically evaluate important aspects of the layout and design of on-screen presentations. The system informs the user about layout-rule violation in a self-explanatory manner, if needed with supplementary visualizations. The paper describes a prototype that checks important general guidelines and standards for effective presentations. We believe that the system exemplifies a high-potential new application area for human-computer interaction and expert-assistance systems.

1 INTRODUCTION

At the onset of their oral presentations, speakers often apologize for the potentially suboptimal quality of the accompanying visual slides¹. They wonder whether the audience can see presented curves at all because the contrast between foreground and background is poor such as yellow on white background. They ask whether the audience in the back can read 12pt fonts well enough and similar questions meant to be rhetorical — the audience often perceives them as cynical.

Why can’t assistant systems inspect slides during writing? In many areas of human-computer interaction, such as web-site design, assistant systems are available nowadays but, to our knowledge, not in the area of audiovisual presentations. The present paper describes a prototype that automatically checks various general guidelines and standards for effective audiovisual presentations.

In our system, short traffic-light inspired bars inform the user about the evaluation result — on demand supplemented by a more elaborate explanation. In the list of preferences, the user can deselect features s/he is not interested in along with personalized values overwriting the system’s defaults. For instance, the slides might become more filled in a lecture than in a business talk. In this paper we focus on the feedback visualization of noticed violations of presentation rules. We illustrate precautions of the system so that novices as well as experts can easily use our system. The implementation of algorithms such as calculating the fullness of a slide or how to detect low contrast is not discussed here.

The paper is organized as follows. In the next section, we sketch the state of the art in assistant systems. In Section 3, we specify important to-be-evaluated criteria in the area of (audio)visual presentation design. The current prototype is discussed in Section 4. In the final section, we draw some conclusions and address future work.

2 STATE OF THE ART IN ASSISTANT SYSTEMS

Automated assistance in user-interface design is a relatively young but dynamic field. The trend in this direction tries to counteract the huge amount of poorly designed interfaces — a phenomenon supported by easy to use tools for implementing a dialogue system. An early seminal attempt constitutes the framework DON (Kim and Foley, 1993), which uses rules from a

¹Although true slide projection is hardly in use anymore, the term slide is still very common for the virtual counterpart of the once physical exemplar.
knowledge base to provide expert assistance in user-dialogue design. It can generate layout variants in a consistent manner. Subsequent development of the assistant systems proceeded in two main directions: graphic art (printing) and web —, and has already given rise to expert-assistant systems with commercial applications.

In the graphic-arts industry, quality control before printing plays a crucial role by reducing the costs of reprinting. The process has been dubbed "pre-flight". In general, the term designates the process of preparing a digital document for final output as print, plate or for export to other digital document formats. The first commercial application was "FlightCheck" described in a paper entitled "Device and method for examining, verifying, correcting and approving electronic documents prior to printing, transmission or recording" Crandall and Marchese (1999). Recent products in the area provide an integrated preflight functionality (see, e.g., Adobe InDesign and Adobe Acrobat). The main objective of these instrument is to reveal possible technical problems of the document. Accordingly, they work with the following primary checklist: (1) Fonts are accessible, compatible and intact; (2) Media formats and resolution are conforming; (3) Inspection of colors (detection of incorrect/spot colors, transparent areas); (4) Page information, margins and document size.

According to Montero, Vanderdonckt & Lozano (2005), the abundance of web pages with poor usability is largely due to shortage of technical experts in the field of web design. In web design, Ivory, Mancoff & Le (2003) present an overview of systems that are capable of analyzing various aspects of the web pages. Historically different browsers have different views on the implementation of web-standards (see, e.g., (Windrum, 2004)) with as a consequence that the same web page may look differently in different web browsers. The above criteria have led to the situation that tools for web-page analysis focus primarily on technical and marketing aspects of the web pages. In general, the term designates the process of preparing a digital document for final output as print, plate or for export to other digital document formats. The first commercial application was "FlightCheck" described in a paper entitled "Device and method for examining, verifying, correcting and approving electronic documents prior to printing, transmission or recording" Crandall and Marchese (1999). Recent products in the area provide an integrated preflight functionality (see, e.g., Adobe InDesign and Adobe Acrobat). The main objective of these instrument is to reveal possible technical problems of the document. Accordingly, they work with the following primary checklist: (1) Fonts are accessible, compatible and intact; (2) Media formats and resolution are conforming; (3) Inspection of colors (detection of incorrect/spot colors, transparent areas); (4) Page information, margins and document size.

According to Montero, Vanderdonckt & Lozano (2005), the abundance of web pages with poor usability is largely due to shortage of technical experts in the field of web design. In web design, Ivory, Mancoff & Le (2003) present an overview of systems that are capable of analyzing various aspects of the web pages. Historically different browsers have different views on the implementation of web-standards (see, e.g., (Windrum, 2004)) with as a consequence that the same web page may look differently in different web browsers. The above criteria have led to the situation that tools for web-page analysis focus primarily on technical and marketing aspects of the web pages. Current web analysis tools primarily check:

- Content, media and script sizes;
- Accessibility of various devices.

Despite the emphasis on purely technical aspects, several publications report on systems assisting users on other aspects of web design (e.g., (Tobar et al., 2008)). Some state-of-the-art systems (see, e.g., (Nagy, 2013)) advise on visible content prioritizing, check the size of control elements (e.g., some controls may be too small for using on mobile devices), and distances between the visible elements of a web page.

An essential question concerns whether or not assistant systems should react directly, in a daemon-like fashion, to any undesirable user action (maybe even forbidding and overruling user actions), or should become active only on demand. The majority of systems mentioned above prefer the on-demand dialogue. Basically, the decision depends on the aspect evaluated. For instance, if the system cannot react to a user action such as saving a file in the current format, the consequence should be brought to the user’s attention. In case of less disastrous effects, the system can react in two manners. According to the first alternative, no ill-formed result can be produced at all (e.g., automatic word correction for typos in SMS typing avoiding unknown words). However, in this mode, the user might feel too much patronized. As a consequence, users tend to switch off such components. Moreover, the second alternative of giving advice on demand offers more freedom to the user (e.g., new words can be typed). In design, violation of rules is used as a stylistic matter (cf. provocative design).

### 3 PRESENTATION RULES

Here we summarize well-known standards for user-interface design in general that also apply to presentation design. Additionally, we list rules of thumb specifically for presentation design in particular. Due to space limitations, we cannot give a comprehensive overview of such rules and standards, and instead focus on the type of rules that our system checks automatically.

Many user-interface design rules (cf. the norm EN ISO 9241) can be applied for a slide presentation as well: use only few different colors; avoid high color saturation levels; give sufficient contrast to the used colors; group related elements together potentially with a frame around them, and/or make sure there is sufficient spacing between non-related items (cf. Gestalt theory; see, e.g. the reprint of original work in (Wertheimer, 2012)); do not make the interface too
crowded; distribute objects such that the virtually assumed grid lines are minimized (i.e. make the interface — in our case, the slides — look balanced and sophisticated; cf. (Galitz, 2007)). The recommendation not to overtax the short-term memory of the user in interface design also holds for a slide: it restricts the number of presented items to $7\pm 2$ per slide (cf. Miller’s rule, (Miller, 1956)). In total, no more than 30% to 40% of a slide’s surface should be occupied.

For consistency reasons (cf. (Shneiderman and Plaisant, 2004)), font, size, position and color of the slides should remain the same in publishing media. This holds in particular for the title. Moreover, the latter’s position should remain the same on each page. Often a predefined frame is assumed for a user interface (cf. the slide master in PowerPoint⁶ for the adaptation to visual presentations).

Furthermore, a wide variety of books focuses on specific rules in visual-presentation design. The books target at different user needs such as presentation for beginners or for professional presenters in business. For instance, for non-designers, Robin Williams (2015) cites four principles of visual presentation design: Contrast, Repetition, Alignment and Proximity.

We focus on the following rules of thumb that, we assume, hold for business presentations. They represent the defaults of our prototype:

1. Do not use more than two font types in a presentation⁷;
2. Do not use fonts smaller than 18pt;
3. Do not use more than three colors;
4. Avoid saturated colors (threshold 30%);
5. Provide sufficient contrast for chosen colors/gray values (threshold 10%);
6. Provide sufficient distance between unrelated objects (as opposed to related objects which should be closer together due to Gestalt theory effects; horizontal = 0.8cm, vertical = 0.8cm⁸);
7. Provide a balanced distribution of elements (maximum number of grid lines = 20 with unified distance of 0.3cm);
8. Slides should not be too full (threshold 30%).

For convenience of the audience, provide automatic print versions without images and/or inversion of a dark background to white with automatic inversion of the foreground colors to black or a user-defined value. Notice that this mode is not discussed in the following for reasons of space.

As will be outlined in the next section, the above mentioned features are first checked per slide according to the default or user-defined parameter settings. The per-slide evaluation reports are subsequently inspected for overall consistency of the entire presentation.

4 SEAP TOOL: A PRESENTATION ASSISTANT SYSTEM

The name SEAP stands for Software-Ergonomic Analysis of Presentations. First, we describe SEAP tool’s system design, e.g., its input and output structure. Then, we focus on the inspection per slide. In Section 4.3, we elaborate on the preferences the user can express for any feature in any particular slide. Section 4.4 indicates how the content of the per-slide evaluation report is used for checking the overall consistency of the presentation.

4.1 System Design

Our prototype is implemented in Java ⁸. As the main input format, we use Portable Document Format (PDF), being the de facto standard for fixed-format electronic documents (cf. ISO 32000-1:2008¹⁰). This means that any presentation can be analyzed that can be exported as PDF, irrespective of the slide presentation program or the operating system with which the presentation was created.

The PDF format also provides access to the presentation’s internal content stored as text, as raster or vector graphics, or as multimedia objects. If available, we use this information for the subsequent slide analyses. However, an analyzed slide may consist of only a picture, without any text information (e.g., when the entire slide is the snapshot of a screen). In this case, or when graphical elements on the slide display text, we use the computer vi-

⁷This task can be extended with a check for whether dispreferred fonts are being used (e.g. Antiqua; for pros and cons of various fonts, see, e.g., (Williams, 2015)). Our default list is based on (Schildt and Kürsteiner, 2006). The user can edit this list (as s/he can any default parameter of the system).
⁸Notice that these values can also be automatically calculated using the font size used in the currently considered box (cf. (Galitz, 1991)).
tion library OpenCV\textsuperscript{11} to identify the objects. Obviously, this variant is computationally more complex and more time consuming which is reflected in lower processing speed especially when producing a report for a larger input file. However, the system gains independence from the actual representation format of the content of the slide. In the following, we do not elaborate on implementation details of the two different methods to obtain an evaluation result. (See (Dünnheber, 2015). This paper also discusses the estimated quality of the evaluation algorithms applied in SEAP.) For the levels of detail we discuss here, it is sufficient to agree that any evaluation result we refer to in the following can automatically be calculated.

Given the decision to inspect a PDF file of the presentation, the way SEAP tool provides the output is also determined. As mentioned in Section 2, an assistant system can evaluate online during the design process, or produce a review on demand. The latter (also SEAP tool’s) has the following advantage of avoiding disturbing the user, especially during stages where the focus is on content rather than form. However, this decision has a drawback: information that would be immediately at hand online (e.g.: Which areas belong to the master slide? Which text box is meant to be the title?) has to be recomputed.

We target different user groups: not only novices but also presentation professionals. Basically, the report aims at easy understandable comments (e.g., in terms of visualizations rather than technical terms in case of novice users). Professionals receive short traffic-light-style comments only.

Moreover, the personal settings for all parameters of the individual evaluation algorithms allow different levels of detail. Inexperienced users see intuitive labels. Professional users can operate an “Advanced” button to enter exact values. (e.g., see Figure 6 in Section 4.3 for the interface enabling personalization of the grid inspection parameters).

In the next paragraph, we delineate the evaluation of an individual page varying according to dedicated user-preferences.

### 4.2 Report Generation per Slide

In the summary of any specific feature in an evaluation report, the green vs. red background indicates compliance or failure of the rules. This traffic-light-style information helps professional users to speed up reading — on the assumption they search for red bars only (cf. Figure 1). It also supports users who are unfamiliar with presentation rules. They can read the traffic light colors as hints whether they are on the right track or not. Moreover, we present informative visualizations whenever possible. If desired, the report can become personalized in two respects:

1. The user has the option to define personal preferences overruling the default settings used in the algorithms check.
2. Additionally, the system offers the choice between short or elaborate report.

![Figure 1: Concise analysis report. The user has asked the system to check font size and crowdedness only: Positive feedback for used fonts is displayed against a green background, negative feedback on crowdedness against a red background.](image1)

![Figure 2: Elaborate font information based on the yellow rule of thumb in the right panel.](image2)

In the following, we focus on the elaborate reporting mode. On each slide, SEAP tool counts the number of different fonts and compares it against the threshold (whose default value is two). It also checks the occurrence of user-defined but generally dispreferred fonts. Figure 2 illustrates the most elaborate version of a font warning generated by SEAP tool. Color saturation warnings and warning for too many different colors on the same page look similar. For reasons of space, we skip details here.

Whenever possible, visualizations are used to inform the user in a self-explanatory manner so that non-professionals can use the system as well. For instance, the system exemplifies whether closely neighboring objects are presumably perceived as belonging together according to the Gestalt laws. The sys-

\textsuperscript{11}OpenCV (Open Source Computer Vision), \url{http://opencv.org} (Nov. 12, 2015).
System transforms such objects into one abstract box in line with the default or user-defined threshold (cf. Figure 3 corresponding to the slide depicted in Figure 1). Notice that, here, the system does not attempt to warn against errors but merely visualizes the most likely grouping perceived by the audience. Therefore, only the user—not the system—can adapt the slide to the intended content. The image also illustrates the difference between PDF-based and image-based inspections. In the PDF file, the two text items are shown in one box (cf. green boxes in the grid representation in Figure 4 according to the predefined settings to highlight text compared to images outlined in Figure 6 in the next subsection). However, given the current threshold setting, an image analysis of the slide would interpret the text items as two independent boxes. Consequently, the user might feel included to improve the slide by positioning the two text items closer together. In SEAP tool, we currently take the PDF information about text to determine text boxes. Thus, no conflict needs to be resolved.

In a similar manner, the information is visualized whether a balanced distribution of objects prevails giving the impression that the user has immersed in the presentation design. A visualization depicts the virtual grid according to a threshold determining which distance is assumed to be one unit. For instance, on the slide in Figure 1, the two images are not fully vertically aligned (cf. Footnote 13). A very exact threshold (e.g., 1 cm) would indicate two vertical grid lines to the left and two vertical grid lines to the right of the images. If the threshold would be set to a more lenient value, there is only one grid line calculated. Figure 4 delineates the result for the case of an exact threshold in order to illustrate the power of the automatic calculation. As holds for all preferences of SEAP tool, the color of boxes and lines displayed in order to highlight the meta information on a slide can be determined by the user to clearly separate the prevailing colors on the slide from the evaluation information added by SEAP tool.

As for contrast evaluation against a given threshold, in the current SEAP tool version, each slide is translated into a grayscale version by applying a black-and-white filter, e.g., a dithering algorithm. The concise report can issue warnings that information has disappeared. Too close color similarities can also be noticed if the threshold is refined. In the elaborate version, slide areas with missing information are highlighted, so that the user does not overlook easily missed details. Currently, we run experiments with the determination of contrast indicated directly on the original page without applying a black-and-white filter. Additionally, a new evaluation rule should be added to the list of evaluated features which applies an algorithm that can recognize colors invisible to colorblind users.

In the next paragraph, we discuss how user-defined preferences are entered. Here, it is important to use terminology that any kind of user understands—not only experts.

In this figure, we use black as the color denoting such boxes because this yields better interpretability of the scaled-down image. In SEAP tool, the user can select any color and any level of transparency.

The obvious grid violation of an exact vertical alignment of the two images is intended here. We use the same image for illustrating the virtual grid calculation later on in the section. At the moment, one can see that the default parameter for grid inspection can be considerably high. Obviously, the original slide as presented in Figure 1 looks balanced.

For an easily understandable and nicely visually supported description see http://www.tannerhelland.com/4660/dithering-eleven-algorithms-source-code/ (Nov. 12, 2015).
4.3 User-specific Preference-Dialogues for Individual Slide Inspection

In this section, we introduce parameter settings for an individualized slide inspection. Notice that the user can tailor slide-specific defaults as well as presentation-general ones. The latter ones are discussed in the separate Section 4.4 because checks of overall consistency deploy the per-slide reports. Moreover, the preference menu offers a separate submenu for the overall presentation parameters. This menu also allows skipping the final overall evaluation if the user is not interested in this inspection, or when s/he is finalizing the presentation.

Invariably, the user can select which features to be evaluated per slide. This can speed up the process considerably\(^\text{15}\). Moreover, the user might be interested in specific feedback only. Thus, s/he gets a list providing the options (1)-(8) presented in Section 3 to select or deselect from. If s/he deselects an item, it becomes gray and moves to the end of the list. This behavior should elicit another user-option available in this window. The user can order the sections of the evaluation report. In the top of the window, the user is informed that the list can be re-ordered if desired. In Figure 5, the window is depicted in the original order. However, the Figure depicts a state where the user has deselected the last five items (cf. gray color). Of course, any choice and order can be revised before applying. Pushing the "Abort" button remains with the previous settings. Pushing the preselected "Apply" button tailors the report according to the user's preferences.

After the user has left the window — irrespective of pushing the "Apply" or "Abort" button, the remaining items provide the choice for a short or a long report variant per feature in the subsequent window. This window provides a flipping choice button where the default is on long as we assume in the beginning that the user, either a novice or even a professional, might like to become used to SEAP tool's feedback behavior. For reasons of space, we have skipped this window here.

Beside the dialogue about the overall order and detailedness of the report, the user can overwrite any default-parameter setting of any feature chosen to be checked in the report. Menu items referring to deselected features for evaluation remain inactive — depicted in gray. We display items always in the same order irrespective to the report order chosen by the user as it is presumably faster to search in a fixed-order menu. Figure 6 shows an example that avoids exact numbers to be changed which is assumed to be the desired mode for novices. The example illustrates how not professional presenters can intuitively work with the SEAP tool. More abstract terms instead of exact values are provided to allow the user to make a meaningful choice. Experts probably prefer a window where they can change the default value directly. The current prototype is not fully capable of different menus for all features. Accordingly, we currently revise and extend these dialogues considerably for the next version.

\(^{15}\)Notice that there exists an invisible part of the report required for the overall presentation checking (see next section). If the user wants many general consistency features to be checked according to the personal preferences, the system can obviously not speed-up.

Figure 5: Personalization of features to be evaluated along with the option to personalize the order of result presented in the report.

Figure 6: Upper half of the dialogue window: Setting of the grid evaluation parameter in the preferences list in a manner that non-professional users can make a meaningful choice; in the lower panel, the button named "Advanced" located above the final choice "Abort/Apply" provides a window with detailed setting options by numbers, preferably used by the experts.

All defaults as outlined in Section 3 can be overwritten. Furthermore, the list of non-accepted fonts can be modified. For reasons of space, we do not elaborate on the fact that there are predefined forbidden values (e.g., zero fonts to be used at all). Of course, the algorithms activated during the evaluation process first check explicitly whether the ranges set by the user are acceptable. Otherwise, the system would un-
expectedly crash.

Based on all these settings, the user gets a review per page in his/her personal style (e.g., a brief traffic-light coding for some selected features only). In its most elaborate mode, the report sums up positive and negative evaluation results for all inspected problems. Additionally, it can provide hints to why/how the slide should be changed.

In the next section, we describe how the evaluation report itself is deployed to detect overall consistency violations.

### 4.4 Evaluation of Overall Consistency

In this section we discuss features that can be checked consistently in all slides such as whether or not the same font has been used throughout the presentation. The user can switch this evaluation on/off in the same manner as for the features to be checked per slide.

Presuming that the user wants a consistency check, in case of a violation, the user gets a resume where to find these cases. The visible and invisible information of the evaluation report per page, enables the system to produce such a report automatically. However, the system needs additional information about the presentation to do a more advanced job.

SEAP tool should know about facts like assumed title position. Basically this is known at design time of the slides but not noticeable in the PDF file SEAP tool uses as input. As for a hypothetical minimal slide master, the system assumes as default a margin area around the presentation of 1cm. This assumption fulfills a general rule of thumb that one should leave some margins all around a slide. A title area is not preset by the system because warnings about any violation would irritate the user who has no idea where the system assumes the title to be — there are no user expectations the system can take for granted. These (minimal) default settings avoid an obligatory dialogue with the user before running the system.

As for any preference in SEAP tool, the user can make changes for these defaults. For better results, the preference menu provides windows that allow tailoring these settings. Here, the user can define areas to remain as uninspected vs. inspected for identity. Figure 7 illustrates the determination of a slide master — an empty frame is assumed on the example slide depicted in Figure 1 — to be identical/ignored over all slides. The individual margin areas can be varied as indicated by the red arrows depicted in the middle of each default margin of 1cm from each side of the slide. The inspection method deploys as default an identity check. We omit the dialogue to select between the options of ignoring the area or to match it throughout the presentation, respectively. This choice window pops up when activating the red arrow of changing an area or when double-clicking on the area. As a consequence, the color of the region changes. Blue means check exactly whereas red means ignore the content completely. First experiments show handiness of the concept in the right vertical periphery — as is depicted as desired personalized setting in Figure 7. This setting allows the user to violate the right margin supposed to be identical with the master slide intentionally due to longer lines.

![Figure 7: User interface determining the master-slide area in the presentation in a dialogue with the user varying the default area to be matched exactly on all slides or to be ignored on all slides. The areas in blue reflect the wish for an exact match and the ones in red, for ignoring any difference in the chosen rectangular.](image)

In a similar interactive window, the area for the title can be specified. Of course, for titles the check is not meant to be an exact match. The user can determine which features should be checked (defaults are font type, font size and color). The same dialogue-window type opens if the user wants to keep more selected areas checked for consistency (e.g., page numbers). In these windows the title of the blue area has to give the region a unique name before applying. The evaluation report refers to warnings for these areas by using the user-defined field names. The same parameterized procedure inspects the title area as well as the user-defined areas (parameters: name, coordinates and features to be checked throughout the presentation). The source of inspection is the fully elaborate form of the evaluation report per slide that SEAP tool yields internally.

In the end of the report, the consistency-check summary is provided to the user. It results from a
final inspection of all internal entries in the evaluation report per page for each feature the user wants to be globally checked. For instance, the system can generate a warning in the final summary such as ‘Attention, on slide 4, the font of the title is inconsistent. Please change from Times to Arial.’.

5 CONCLUSIONS

In this paper, we have delineated a prototypical assistant system for the evaluation of visual presentations. We have illustrated the diversity of topics automatically checked by our system. So far, no such tool is available on the market. Given the often poor quality of presentations in science and business, such a system for automatic layout and design evaluation of slides is highly desirable.

Our system, can automatically evaluate visual presentations according to well-known rules for designing any kind of user interface as well as some specific presentation rules. Our system reads-in a PDF file of the presentation so that the system is independent from the way the user obtained the presentation. The system performs different inspections on the PDF file as well as some analyses of an image representation of each slide. Based on these results, the SEAP tool delivers an evaluation report per slide in a personalized manner. Furthermore, the user can determine which features are evaluated at all and in which order the results are presented. In addition, individual parameters for the evaluation calculation can be personalized along with the level of detail of the report. If the user wants, general consistency throughout the slides can be evaluated as well at the end of the report.

As for future work, we plan to translate new rules of presentation design and layout into automatic evaluation procedures. For instance, as outlined in Section 4.2, color-blind proof-reading of slides should be available. Furthermore, a deeper image analysis should be imposed on the system. As mentioned in the previous section, many existing analysis components of SEAP tool can be improved. Moreover, new checks can be added based on the existing components. Additionally, user studies should be conducted to test the user interface of the system with specific target groups like novices and professional users, respectively. We paid attention to the fact that the dialogues become clear even to novices. In this regard, we supported text with intuitive visualizations. However, only a study can provide clear insights into an optimal user interface. Moreover, we would like to ask the users what is their ranked list of features they would desire most.

ACKNOWLEDGEMENTS

We grateful to Gerard Kempen for his valuable comments on earlier versions of the paper.

REFERENCES


