



HAL
open science

Reification of visual properties for composition tasks

Ghita Jalal

► **To cite this version:**

Ghita Jalal. Reification of visual properties for composition tasks. Human-Computer Interaction [cs.HC]. Université Paris Saclay (COmUE), 2016. English. NNT : 2016SACLS554 . tel-01599253

HAL Id: tel-01599253

<https://theses.hal.science/tel-01599253v1>

Submitted on 2 Oct 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

NNT : 2016SACLS554

THÈSE DE DOCTORAT
DE
L'UNIVERSITÉ PARIS-SACLAY
PRÉPARÉE À
L'UNIVERSITÉ PARIS-SUD

ECOLE DOCTORALE N° 580
Sciences et Technologies de l'Information et de la Communication

Spécialité de doctorat : Informatique

Par

Mme. Ghita JALAL

Réification des propriétés visuelles pour les tâches de composition

Thèse présentée et soutenue à Gif-sur-Yvette, le 16 Décembre 2016.

Composition du Jury :

M. HALSKOV Kim	Professeur à l'université d'Aarhus	Rapporteur
M. PRIÉ Yannick	Chercheur à l'université de Nantes	Rapporteur
Mme. GENTES Annie	Maître de conférences à Télécom Paris Tech	Examineur
M. FEKETE Jean-Daniel	Directeur de recherche à Inria Saclay	Président du jury
M. CHATTY Stéphane	Chercheur à l'école nationale d'aviation civile	Examineur
Mme. MACKAY Wendy	Directeur de recherche à Inria Saclay	Directeur de thèse

To my mother.

ACKNOWLEDGEMENTS

This work would not have been possible without the tremendous contribution of all designers, artists, scientists and engineers, who participated in our studies. I would like to thank all of you for your time, patience, and insights. Your stories are the pounding heart of this work. Your stories kept me focused, grounded, and going. I would like to thank the reviewers and the committee members for their feedback on an earlier version of this thesis. Wendy, thank you for teaching me to always put users first.

I would like to thank everyone in the ex-situ team, for the interesting discussions, comments and feedback throughout my thesis. Wendy and Michel, thank you for the warm welcome to the lab. The theoretical discussions during my internship encouraged me to start a PhD. Cédric, Fanis and Sarah, thank you for your guidance. Clemens and Joanna, thank you for the inspiring exchanges on projects that shaped this work. Nolwenn, thank you for the great collaboration. Alexandra, Alex, Nacho, Carla, German, Mai, Sally, Mickael, Yujiro, Philip, Stacy, Abby, and to all interns, and other members of the team, Thank you all for the good spirits, and for making working in the lab enjoyable.

Marianne, Lydie, Céline and Thalia, your presence throughout the past three years gave them taste. I would not trade the times we spent together for anything in the world. Thank you for being there till the end.

Ghada, Papa, Mama, I cannot thank you enough for making me who I am today.

CONTENTS

1	INTRODUCTION	1
1.1	Illustration: Visual composition practices	2
1.2	Thesis Statement	4
1.3	Research Approach	4
1.3.1	Critical Object Interviews	5
1.3.2	Grounded Theory	6
1.3.3	Comparative Structured Observation	6
1.4	Thesis Contributions	7
1.5	Thesis Overview	8
2	CONCEPTUAL FOUNDATIONS	11
2.1	Concepts in this thesis	11
2.1.1	Visual Properties	11
2.1.2	Designers' visual composition practices	12
2.2	Theoretical framework	12
2.2.1	Instrumental Interaction	12
2.2.2	Framework for reifying visual properties	13
2.3	Perceiving visual properties	14
2.3.1	Perceptual Organization	15
2.3.2	Affordances and Perceptual learning	17
2.4	Visual composition guidelines	18
2.5	Representing visual properties	21
2.5.1	Color	22
2.5.2	Typography	33
2.5.3	Layout and Visual Organization	35
2.6	The Visual composition process	39
2.6.1	Visual composition is a reflective practice	39
2.6.2	Manipulating visual properties in situ	40
2.7	Summary	41
3	RELATED WORK	43
3.1	Direct manipulation	43
3.1.1	<i>Sketchpad</i>	43
3.1.2	The <i>Dynabook</i>	45
3.1.3	The <i>Xerox Star</i>	46
3.1.4	Direct Manipulation Principles	47
3.2	The desktop metaphor	47
3.3	Visual composition tools	49
3.3.1	Visual composition tools for novices	49
3.3.2	Visual composition tools for experts	54
3.4	Visual composition tools support of the reification of visual properties	60
3.5	Summary	62

4	MANIPULATING VISUAL PROPERTIES IN A DESIGN APPLICATION	63
4.1	Introduction	63
4.2	Context	65
4.2.1	Designers's composition practices	65
4.2.2	Approaches to study software use	65
4.3	Initial Observations	67
4.4	Study of designers tools and strategies for manipulating visual properties	68
4.4.1	Participants	69
4.4.2	Hardware and Software	69
4.4.3	Task	70
4.4.4	Procedure	70
4.4.5	Data Analysis	71
4.4.6	Results	71
4.4.7	Strategies to manipulate visual properties: Clever Tricks and Bad Hacks	76
4.5	Interview Study: How common are Clever Tricks and Bad Hacks?	78
4.5.1	Participants	78
4.5.2	Procedure	78
4.5.3	Results	78
4.6	Discussion	80
4.6.1	Appropriating visual properties	80
4.6.2	Implications for design	81
4.7	Summary	83
4.8	Contributions	83
5	ARTISTS AND DESIGNERS COLOR MANIPULATION PRACTICES	85
5.1	Introduction	85
5.2	Context	86
5.3	Participants	86
5.4	Procedure	87
5.5	Data Collection	87
5.6	Data Analysis	88
5.7	Results and Discussion	89
5.7.1	Personal color representations	89
5.7.2	Manipulating color in context	93
5.7.3	Manipulating personal color representations in context	96
5.7.4	Implications for design	97
5.8	Summary	97
5.9	Contributions	98
6	GRAPHIC DESIGNERS' LAYOUT PRACTICES	99
6.1	Introduction	99
6.2	Context	99

6.3	Participants	100
6.4	Procedure	100
6.4.1	Data Collection	100
6.4.2	Data Analysis	101
6.5	Results	101
6.5.1	Manipulating links between spatial properties	101
6.5.2	Flexible and fixed visual properties	103
6.5.3	Guidelines for dynamically changing visual properties	105
6.5.4	Changing the digital nature of visual elements to access specific visual properties	105
6.6	Discussion	106
6.6.1	Beyond composition factors: flexible visual properties for dynamic layouts	107
6.6.2	Implications for design	107
6.7	Summary	108
6.8	Contributions	108
7	DESIGNING FOR MANIPULATING VISUAL PROPERTIES	111
7.1	Manipulating visual properties: Implications for design	111
7.2	Manipulating visual properties: interactive prototypes	113
7.2.1	Color Prototypes	113
7.2.2	Study: Using Color Prototypes	117
7.2.3	Manipulating visual properties to structure visual content	119
7.2.4	Study: Using Layout prototypes	122
7.3	Summary	124
7.4	Contributions	124
8	CONCLUSION	127
8.1	Thesis Summary	127
8.2	Contributions	128
8.2.1	Theoretical perspectives: Framework for reifying visual properties for composition	128
8.2.2	Empirical Findings: Designers' composition practices	129
8.2.3	Technologies: Designing for the manipulation of visual properties for composition tasks	129
8.3	Limitations and perspectives	130
A	CRITICAL OBJECT INTERVIEWS: COLLECTED DATA	133
B	COLOR PORTRAITS: INTERVIEW QUESTIONS	135
B.1	Interview objectives	135
B.2	Interviewee recruitment	135
B.3	Interview proceeding	135
B.4	Debriefing	135
C	GROUNDED THEORY CATEGORIES	137
C.1	Typologies of Structures:	137
C.1.1	Semantic structures:	137

C.1.2	Spatial structures:	137
C.1.3	Pattern structures:	137
C.1.4	Conceptual structures:	137
C.1.5	Example structures:	137
C.2	Interaction with Structures:	138
C.2.1	Reuse/Application:	138
C.2.2	User-System Partnership:	138
C.2.3	Adaptability:	138
C.2.4	Informing Content:	138
C.2.5	Filtering	138

BIBLIOGRAPHY	139
--------------	-----

LIST OF FIGURES

Figure 1	Creating personal color representations	3
Figure 2	Thesis process. My research includes a variety of methods that include observing designers in field and laboratory settings, designing a variety of interactive technologies to support manipulating visual properties, connected with a common theoretical foundation.	5
Figure 3	Perceiving visual properties.	12
Figure 4	Framework for the reification of visual properties for composition tasks	13
Figure 5	Top row shows how novel features emerge as additional dots are added to a stimulus, while the bottom row shows the same for line segments.(Pomerantz and Cragin 2015)	16
Figure 6	Monochromatic picture uses color to reinforce unity. Damon Winter. Personal photograph from Iceland. Communication Arts, May/June 2005. (Lauer and Pentak 2012)	19
Figure 7	Geometric unity with variations on a theme. Charley Harper. Original painting for Titmouse Tidbit (Tufted Titmouse). 1975. Acrylic on canvas, 1' 1" × 1' 2". (Lauer and Pentak 2012)	20
Figure 8	All aspects of compositions attract attention to the grapefruit as a focal point. Susan Jane Walp. Grapefruit with Black Ribbons. 2000. Oil on linen, 8" × 8¼". Tibor de Nagy Gallery, New York. (Lauer and Pentak 2012)	21
Figure 9	An unbalanced design creates uneasiness for the viewer. (Lauer and Pentak 2012)	21
Figure 10	Design guidelines suggest changing specific properties to apply the principles of unity, variety, emphasis and balance.	21
Figure 11	Newton's drawing of his Prism Experiment (Wandell 1995)	23
Figure 12	The 12 hues color cycle (p.31)(Itten 1970)	25
Figure 13	Goethe's Colorific Circle (Plate.1 Figure.3)(Goethe 1970)	26
Figure 14	Newton's color circle	26
Figure 15	The Equilateral Triangle p.191 (Albers 1963)	27
Figure 16	Munsell Color Representations	28

Figure 17	Munsell Soil Color Charts (Edward R. Landa and Mark D. Fairchild 2005)	29
Figure 18	RGB and CMY color models	29
Figure 19	The RGB and CMYK gamuts (OKI Data 2016)	30
Figure 20	The relativity of color (p.76)(Albers 1963) . . .	32
Figure 21	Kandinsky’s concentric circles	33
Figure 22	Text size Georgia h as a bitmap and as an outline. (Carter et al. 2015)	35
Figure 23	Changing orientation depends on the frame of reference. (Arnheim 1974)	37
Figure 24	Zürich Tonhalle Musica Viva Concert Poster, 1958, Joseph Müller-Brockmann	38
Figure 25	Petra changes the initial blueprint based on two visual properties: scale and size. (Schon 1992)	40
Figure 26	Ivan Sutherland using <i>Sketchpad</i> (Sutherland 1964)	44
Figure 27	Three digits representing one scalar value in <i>Sketchpad</i>	44
Figure 28	One of the first painting tools designed and implemented in Smalltalk by a twelve-year-old girl (Kay and Goldberg 1977)	45
Figure 29	Progressive disclosure: The <i>Xerox Star</i> ’s property sheets display object properties upon user request (Johnson et al. 1989)	46
Figure 30	Two Interfaces based on the desktop metaphor, centered on the concept of a document	49
Figure 31	DesignScape: The system generates layouts in a variety of styles by sampling a low-dimensional parameter space learned from examples. (O’Donovan, Agarwala, and Hertzmann 2015)	51
Figure 32	Toolglasses and Magic Lenses (Bier et al. 1993)	52
Figure 33	Vector drawing editor using surrogate interaction (Javed et al. 2011)	53
Figure 34	a. A collection of attribute cards, connected to an object. b. Free floating attribute cards (Xia et al. 2016)	53
Figure 35	The Parallel Pies tool can be rotated and moved to selectively show all or part of a variation. A paddle extending from the hub controls the angle of rotation. (Terry et al. 2004)	55
Figure 36	<i>TextDraw</i> and <i>Typels</i> use gestures to draw with type (Kirton 2011)	55
Figure 37	The user drags the “width: 280” shared value and drops it on the “shape: circle” shared value. All circles in the selection now have a width set to 280. (Hoarau and Conversy 2012)	56

Figure 38	Creating links between visual properties (Hou- raou and Conversy 2012)	57
Figure 39	Artists can create links between elements' vi- sual properties using a relational graph (Kazi et al. 2014b)	57
Figure 40	Aquamarine shows support for the history of visual property manipulations (Myers et al. 2015)	58
Figure 41	(a)the image is duplicated and its pixels rear- ranged into a lightness histogram; (b) bright pixels are selected with the rubber-band selec- tion tool; (c) all pixels are rearranged into a hue histogram and yellow pixels are filtered out ; (d) the sky is enhanced by resizing the se- lection on the saturation histogram. (Chevalier, Dragicevic, and Hurter 2012)	59
Figure 42	Composition tools' support of the reification of visual properties.	62
Figure 43	Property Sheets in the Xerox Star and in Adobe Illustrator Cs6	64
Figure 44	Three visual composition strategies for the same visual composition task.	68
Figure 45	Initial and final versions of the two study posters 69	
Figure 46	Number of different strategies per visual com- position task	72
Figure 47	Modular repetition is a reasonable trade off be- tween single action and repeated actions. . . .	73
Figure 48	Categories of tools participants used to com- plete visual composition tasks in the study. . .	74
Figure 49	Tool choices for tasks supported with an appli- cation specific tool.	75
Figure 50	Using a default pie chart, hidden with a circu- lar shape, to simulate menus of an interactive prototype in Keynote.	79
Figure 51	Mapping visual properties and visual compo- sition tasks they were used to complete	80
Figure 52	Two different representations to access color in Adobe Illustrator CS6.	82
Figure 53	Today's color pickers have changed little since 1990.	86
Figure 54	Eight artists and designers demonstrated how they manipulate color to achieve effects in both physical and digital media.	87
Figure 55	Color Portrait illustrating a Critical Object Story.	88
Figure 56	Color Portraits help comparing participants' color manipulation stories.	89

Figure 57	Users pick colors from diverse sources, and often tweak them later.	90
Figure 58	Artists and designers use color representations to reveal useful information about interim states in their work process.	91
Figure 59	Users record color sources and targets, as well as intermediate steps	92
Figure 60	Artists and designers manipulate the visual properties of color swatches within a palette.	93
Figure 61	Changing sizes and positions of colored objects creates different palettes.	94
Figure 62	Artists and designers create composites, by combining color and other visual properties. They manipulate a composite as a single element. They also decompose it to extract composing visual properties.	95
Figure 63	P2 combined color and texture into a single visual unit for manipulation.	96
Figure 64	P5 changed text orientation, while remaining within the grid.	102
Figure 65	P11 used semantic properties of the content to create a link between visual properties.	103
Figure 66	P6 defined fixed and flexible visual properties while structuring visual elements within a poster.	104
Figure 67	P9 programmatically defined the scrolling behavior for navigating a web page.	106
Figure 68	P4 transform text to an image to control the position of each letter individually.	107
Figure 69	Color Partner: Users specify initial colors and move the cursor to control generation of new colors. They can save interesting intermediate colors by clicking on the dot.	114
Figure 70	<i>Color Revealer</i> : Users' hesitations and corrections are revealed through changes in hue and intensity.	115
Figure 71	<i>Palette Explorer</i> : Users create interactive color palettes with dynamic swatches. They can manipulate the visual properties of color swatches in context.	116
Figure 72	<i>Color Compositor</i> : Users combine colors and textures to manipulate composites.	116
Figure 73	<i>Color Compositor</i> : Users combine colors and textures to manipulate composites.	120
Figure 74	<i>Color Compositor</i> : Users combine colors and textures to manipulate composites.	121

Figure 75 *Color Compositor*: Users combine colors and textures to manipulate composites. 122

Figure 76 Photo of a participant’s tools and artifacts in her workspace. 133

INTRODUCTION

Designers use composition tools to create documents such as books, posters, and flyers. In this type of tasks, designers focus on organizing visual content. They define a hierarchy of elements and change their properties to serve the visual message they want to communicate to their viewers. Designers choose color schemes and typefaces. They align graphical objects or distribute them in the document. These tasks rely on manipulating object properties including color, typography, alignment, distribution, size, position, and layering.

Early commercial composition tools such as the Xerox Star (Johnson et al. 1989), the Lisa user interface (Ludolph and Perkins 1998), and the Apple Macintosh (Williams 1984) used direct manipulation principles for graphical objects. In the Star and Lisa systems, *“text, graphics, tables, and mathematical formulae are all edited inside documents”* (Johnson et al. 1989). This model already selected text, graphics, tables, and mathematical formulae as targets for user manipulations. The Xerox Star, the Apple Lisa, and the Apple Macintosh were all based on the principle of progressive disclosure: *“detail is hidden from users until they ask for it”* (Johnson et al. 1989). Thus, elements other than these targets, including visual properties such as color, size, height, position and typefaces, were considered details the user can access through dialog boxes, only when asked for. Progressive disclosure alleviated the problem of overwhelming users with choices and commands irrelevant to what the user is trying to do (Johnson et al. 1989). In this thesis, I am questioning whether visual properties are really irrelevant for composition tasks. Are these properties details for designers trying to organize visual content on a document? Should visual composition tools still hide these properties in dialog boxes similar to the Xerox Star Property Sheets?

Progressive disclosure still prevails in today’s application-based desktop interfaces. More recent applications such as the Adobe Creative Suite, and Microsoft Office, still provide access to properties through menus and dialog boxes. These applications also introduced new tools to manipulate visual properties. The color picker supports selecting a color from the representation of a color space. Guides, grids, and masters, define the position of visual elements in a document. Property inspectors such as the eyedropper tool, extract and copy an object property, and apply it to another object. Designers use desktop interface tools to create their visual compositions. They ma-

nipulate visual properties such as color, size, height, position, and layering of graphical elements.

In this thesis, I focus on designers' tools and practices in visual composition: How do designers manipulate properties when they create their artifacts? How do current visual composition tools support the manipulation of visual properties? And how can we better support designers in creating visual compositions?

1.1 ILLUSTRATION: VISUAL COMPOSITION PRACTICES

Artists and designers manipulate visual properties such as color, texture, size and position, as they create physical and digital artifacts. To understand how designers manipulate color in digital documents, we first look at their practices when they choose colors for physical artifacts. Then, we describe these practices in digital composition tools.

Alice is a ceramist. The following scenario illustrates how she manipulates a specific property: color, for physical and digital composition tasks.

In her atelier, Alice closely follows all the steps from making initial shapes, to firing and polishing the final object. She wants to create red tiles for a kitchen wall. First, she looks at small tiles she used in the past (Fig. 1). For each of these tile colors, she created her own personal code. This code starts with the first one or two letters of the color name, R for "*Rouge*", and then the year where she created the color, 16 for 2016 and the number of trials she went through before getting this specific color. Thus, while creating a red tile, the color resulting from the third trial would have the code: R1603. In a separate notebook, she keeps a table with the color codes, and the corresponding recipe for preparing this color, with quantities in grams of the materials she had used, firing time and temperature. In this notebook, she also writes down any names she gave to this color, and whether she used it as a final color for an artifact. Having access to all this information helps Alice choose a starting color. She knows at this point that she will probably not use the exact same red for her new color tiles. But she picks one recipe as a starting point for a new set of trials.

CREATING PERSONAL REPRESENTATIONS:

Precise color names help Alice remember how she created each color. The code is often linked to the color source, to where she applied it, or to some specific incident while preparing it. She can also recall

these incidents by looking at the trials she went through to get the final color. She compares the recipes, and instantly notices the differences between the two colors. She can also deduce, from looking at the physical tiles, impacts of firing time and temperature.

Alice creates her own color representations. When she wants to select a color for a new tile, she starts from a previous tile, from a notebook with precise recipes of her previous colors, or from physical samples of colors she created (Fig. 1). These personal color representations contain information about the materials she used, the proportions of these materials, and the exact effect they will produce.

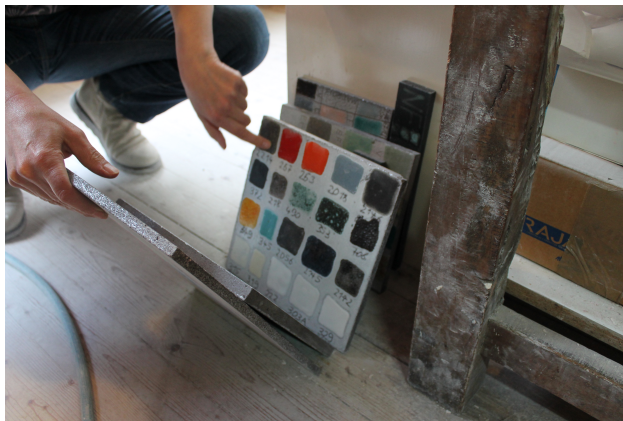


Figure 1: Creating personal color representations

A few weeks later, Alice is invited to present her work in an art exhibition. The organizers sent her their poster and said she could use it to send invitations for the event. Alice wanted to personalize the poster to create small foldable flyers for clients and visitors. Alice first opened the poster in Adobe Photoshop. She wanted to keep the same poster design. She also wanted to maintain the event logo, but change its colors.

FIXED AND FLEXIBLE VISUAL PROPERTIES:

Alice defined the positions and sizes of graphical elements as fixed. The logo colors are flexible, she wanted to use them to create a variation of the initial poster.

Alice picked from her ceramic tiles collection, a set of colors she wanted to use in the flyer. While several tiles had similar colors, they all had different textures. To capture these differences, Alice scanned the physical tiles. She

saved them as images, and opened them within the Photoshop document. She then used the images to color specific areas within the poster. She composed the titles to create a color arrangement she liked.

COMBINING MULTIPLE VISUAL PROPERTIES:

Alice was not interested only in changing the colors. She wanted to create a visual effect through manipulating the combination of color and texture for each scanned tile. Through scanning each physical tile, she wanted to add an independent visual element in her composition: the result of combining color and texture.

This scenario illustrates how Alice finds innovative ways of manipulating visual properties as she creates physical and digital artifacts. Alice uses personal color representations to choose her colors. She defines fixed and flexible visual properties across related compositions. She also combines several visual properties into independent visual elements.

1.2 THESIS STATEMENT

This work supports the following thesis:

Designers use visual properties such as color, size, position and layering, to organize their content. I argue that visual composition tools should reify visual properties—that is, create them into first-class, interactive objects. Visual composition tools should support creating personal representations of these properties. They should support interacting with these properties directly in the composition. They should also support linking properties in a document. Finally, they should support creating rules to define how these properties change in space, in the same composition, or in time, across different compositions.

1.3 RESEARCH APPROACH

Three types of contributions ground this thesis work: empirical findings, technological proofs and theoretical perspectives. This section presents triangulation (Mackay and Fayard 1997) as a method to link these three types of findings. Then, it presents critical object interviews (Mackay 2002b) and grounded theory (Corbin and Strauss 2007) as an analytical method to justify empirical findings, and comparative structured observation (Mackay 2014) (Garcia et al. 2014) as a method

to facilitate systematic comparisons (Fig. 2).

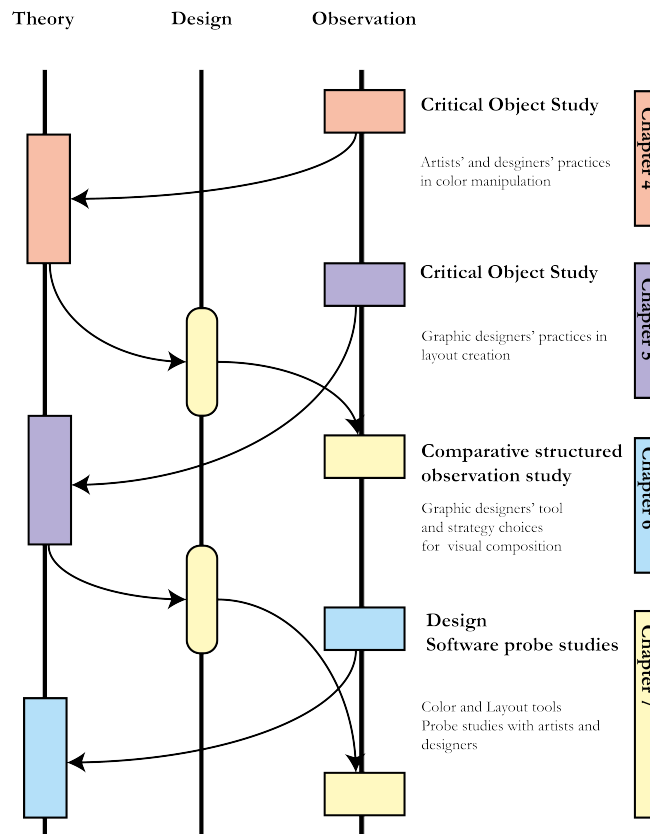


Figure 2: Thesis process. My research includes a variety of methods that include observing designers in field and laboratory settings, designing a variety of interactive technologies to support manipulating visual properties, connected with a common theoretical foundation.

1.3.1 Critical Object Interviews

We use a variation of the critical incident technique (Flanagan 1954) to conduct *critical object interviews* grounded in participants' practices in real projects they are currently working on (Mackay 2002b). We ask participants to show recent artifacts or projects they created. We probe for details about the specific steps participants went through to create this artifact.

We use this interview technique to collect *Critical Object Stories*: each participant's interview includes multiple stories according to the number of artifacts shown and the granularity of the steps described.

We create a visual representation for each critical object story. We call

these representations *StoryPortraits* (Jalal, Maudet, and Mackay 2015); they capture the situated nature of the story-based data we collected from the interviews.

We extract these stories from recordings of heterogeneous formats (Appendix A).

StoryPortraits represent the components of a critical object story in a single-page format. We used StoryPortraits to compare and classify critical object stories. Chapter 5 and Chapter 6 introduce Color Portraits and Layout Portraits; two examples of StoryPortraits, adapted to the stories extracted from two interview studies.

1.3.2 *Grounded Theory*

We use Grounded theory (Corbin and Strauss 2007) to uncover artists' and designers' practices in their recent projects. Grounded theory "denotes theoretical constructs from qualitative analysis of data" (Corbin and Strauss 2007). Grounded theory provides the tools for using empirical field data to let a phenomenon emerge instead of evaluating theories with empirical observations.

Grounded Theory provides a set of methods and strategies for describing and analyzing field observations. Different techniques, including interviews, documents and observations, provide heterogeneous data which are iteratively coded. The codes are then sorted, critically analyzed, compared amongst themselves and grouped into categories. Finally, categories are analyzed by looking at relationships emerging from their descriptions. Grounded Theory does not aim to validate theories by illustrating phenomena, but rather highlights emerging phenomena which can be further explored by various studies. We use Grounded Theory methods to codify and categorize stories extracted from interview studies in chapter 5 and chapter 6.

1.3.3 *Comparative Structured Observation*

We use Comparative Structured observation to compare designers' strategy and tool choices for similar visual composition tasks. Comparative Structured observation (Garcia et al. 2014) (Mackay 2014) combines elements of controlled experiments to allow comparisons and a realistic task to enhance external validity. In contrast to a controlled experiment, the goal is not to assess a clearly defined hypothesis with the data collection. Instead, we want to create comparable conditions for observing and comparing common patterns and processes.

1.4 THESIS CONTRIBUTIONS

In this dissertation, I discuss how the reification of visual properties into first class interactive objects, can provide a better support for visual composition tasks. The thesis offers the following empirical, theoretical and technical contributions:

EMPIRICAL FINDINGS:

- Artists and designers create their own color representations. They start from these representations to select colors for their new artifacts.
- Color manipulation occurs through manipulating the visual properties of color, in the context of its surrounding visual elements in the composition. Artists and designers experiment with the size, position and layering of color swatches. They also combine color with other visual properties such as texture and light reflection, and manipulate the combination as an independent visual element in the composition.
- Graphic designers create links between spatial properties such as size, position and layering. They manipulate these links to create a layout concept.
- Graphic designers define fixed and flexible visual properties. They create rules to define how flexible visual properties change in space, within the same composition, or in time, across related compositions.
- Graphic designers choose strategies where they directly manipulate visual properties on the document. They prefer tools that provide direct access to visual properties, to dialog box tools. They find novel ways to use properties to complete visual composition tasks.

THEORETICAL PERSPECTIVES:

- The framework for reifying visual properties introduces dimensions for the manipulation of visual properties in composition tools. These tools should support creating personal representations of visual properties. They should support manipulation of these properties in the composition. They should also support creating links among these properties. Finally, they should support creating rules to define how properties change in space, across instances of the same composition, or in time, across related compositions.

INNOVATIVE TECHNOLOGIES:

- *Color Partner* supports creating personal, informative representations of the color manipulation process. Users can benefit from the cues the system provides (size of the dots, distance between the dots), to guide their color manipulation process.
- *Color Revealer* permits creating personal, informative color representations of how a process evolves. These representations are based on the progress of an activity over time. They contain information about the length of this activity and visually communicate its phases.
- *Palette Explorer* supports manipulating visual properties of color, such as size, position and layering. Users can change these properties in the context of the surrounding colors. Through manipulating these properties, they can create personal color representations using specific values for color visual properties.
- *Color Compositor* combines colors and textures. It supports manipulating the visual properties of this combination. Users can create this combination visually on the colors and textures of interest. *Color Compositor* also supports extracting color and texture from a colored-texture, and getting its component elements.
- *Linkify* supports linking visual properties directly on the document. It supports visually defining rules for how these links evolve as the content changes.
- *Contextify* supports creating rules that define how visual properties change in a visual composition. It also supports creating these rules directly on the content elements.

1.5 THESIS OVERVIEW

Chapter 2 introduces the main concepts used in this thesis. First, it presents a framework for refiying visual properties for visual composition. Then, it presents factors affecting how designers create their visual compositions. Perception, design tools and guidelines affect how designers create their visual compositions. Then, it describes visual composition as a reflective practice, where designers know more than they say, and respond to the situated circumstances of creating their compositions.

Chapter 3 describes user interface tools that support visual composition. It presents tools that address the manipulation of visual properties through hiding complexity, or automatizing part of the manipulation process. It also presents selected tools that address specific groups of experts, such as artists, illustrators and graphic designers,

to support specific visual composition tasks. Finally, it describes the advantages and remaining challenges of the approaches these tools used to support manipulating visual properties.

Chapter 4 focuses on designer's visual composition practices within a specific design application. It presents a comparative structured observation study of graphic designers, as they perform visual composition tasks in a graphic design application. It extracts and compares the tools and strategies they use to complete visual composition tasks. It presents recurrent strategies across graphic designers and the types of tools they use. Based on this comparison, it suggests new direction for the design of visual composition tools.

Chapter 5 focuses on designers' and artists' color manipulation practices, as they create their artifacts. It extracts and analyzes color manipulation stories from interviews with artists and designers. Based on these observations, it summarizes main color manipulation practices displayed in their projects, and the resulting implications for the design of color manipulation tools.

Chapter 6 focuses on graphic designers' layout practices in their current projects. It extracts and analyzes stories from an interview study with graphic designers. Based on these observations, it summarizes main layout creation practices across graphic designers, and suggests new directions for the design of layout creation tools.

Chapter 7 illustrates how the framework dimensions could be articulated in the form of visual composition tools. It introduces two groups of prototypes. Four color tools demonstrate new possibilities for color manipulation. These tools support creating personal representations, manipulating the visual properties of color, and combining color with other visual properties. Two layout tools demonstrate new directions for layout tools. They support creating links between spatial properties such as size and position, and defining rules for dynamic properties. Two studies present how designers used and described the tools with respect to the practices they support.

CONCEPTUAL FOUNDATIONS

This chapter introduces concepts used throughout this thesis. It presents a framework for the reification of visual properties in visual composition tools, to turn them into first class interactive objects. It then presents the factors that affect how designers create their compositions. First, designers' perception and interpretation of visual stimuli affect how they create their compositions. Also, art and design principles help designers plan and justify their composition choices. These choices also depend on the existing representations designers use to create their compositions. I focus on the representations of color, layout and typography. Finally, visual composition is a reflective process. It depends on the designer's reaction to the current situation. Plans based on rules and best practices guide, but do not completely shape the final composition. For this reason, in situ observations of designers at work is helpful to uncover their composition practices.

2.1 CONCEPTS IN THIS THESIS

This section defines the concepts and theoretical foundations of this thesis. First, the concept of visual properties, also called “attributes”, “features”, or just “properties” appears in several domains. I identify different types of visual properties based on how we perceive and interpret visual stimuli.

Then, I draw from the empirical findings I present in the following chapters, the perspectives explained in this chapter, and the analysis of current visual composition tools to present a framework for manipulating visual properties. This framework describes the main dimensions composition tools should consider to support the manipulation of visual properties for designers.

2.1.1 *Visual Properties*

The Gestalt's concept of emergent features (Pomerantz and Cragin 2015) distinguishes between two types of visual properties. Properties of a component depend only on this component. Emergent features are visual properties created from a particular organization of visual objects. Gibson's ecological view (Gibson 1979) presents visual properties as perceived affordances. These are possibilities for action that depend on the observer's current action capabilities. Perceptual learning (Adolph and Kretch 2015) introduces distinctive features. These are the minimal visual properties required to differentiate a stimu-

lus (Fig. 3).

Visual properties affect how observers perceive objects and the possi-

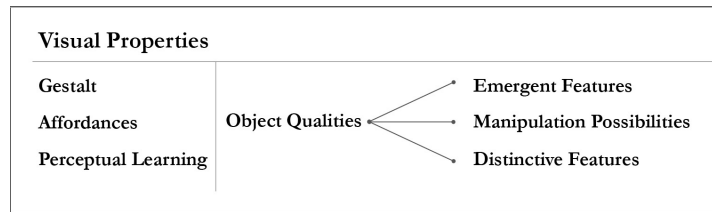


Figure 3: Perceiving visual properties.

ble actions to take on these objects. For this reason, visual properties affect how designers perceive and act on objects as they create their compositions.

2.1.2 Designers' visual composition practices

As they create visual compositions, designers move between stages of problem setting and problem solving (Schon 1984). In this thesis, I focus on the practices designers develop when creating visual compositions. We observe these practices in tasks where designers do not recreate existing artifacts, following detailed procedural instructions. They work instead within a set of predefined initial constraints.

2.2 THEORETICAL FRAMEWORK

In this thesis, I argue that visual composition tools should treat properties as first class objects for interaction. Designers should be able to directly access these properties in the composition. I use the instrumental interaction model, and specifically the design principles of reification, polymorphism and reuse to describe how visual composition tools can transform visual properties into first class objects.

2.2.1 Instrumental Interaction

Instrumental interaction (Beaudouin-Lafon 2000) generalizes and operationalizes direct manipulation principles, suggesting a reification of commands into interaction instruments (Beaudouin-Lafon and Mackay 2000). Domain objects in the instrumental interaction model form the purpose of the interaction. These objects have attributes the user can act on, and can become domain objects of their own.

Two design principles complement reification. Polymorphism means that one command can be applied to objects of different types. Reuse

makes users' previous input, output or both available in the current context (Beaudouin-Lafon and Mackay 2000).

In this thesis, I propose the reification of attributes, visual properties of domain objects. These properties become first-class objects for interaction. I argue that the reification of visual properties into first-class objects, provides a better support for designers in visual composition tasks.

The reification of properties in a visual composition tool creates new objects of interest. Polymorphism helps to maintain a small number of commands, as the number of objects of interest increases. Thus, the same polymorphic commands, can operate on new reified objects, such as visual properties. Reification of properties will also facilitate output reuse, since new objects will be available for the user's upcoming visual composition tasks.

2.2.2 Framework for reifying visual properties

We use the empirical findings from observing designers (Chapter 4, Chapter 5, and Chapter 6) to create a framework for the manipulation of visual properties in visual composition tasks (Fig. 4). This framework is based on the design principles of reification, polymorphism, and reuse.

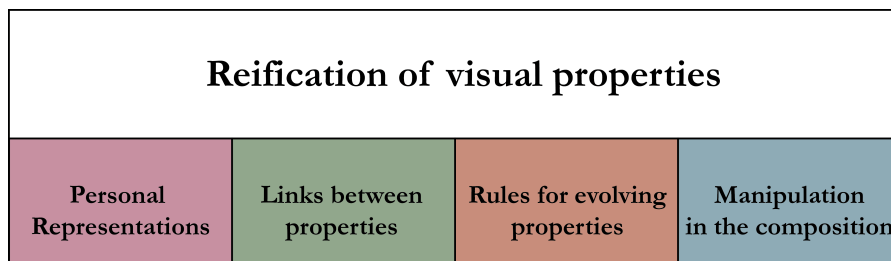


Figure 4: Framework for the reification of visual properties for composition tasks

I argue that the reification of visual properties into first-class interactive objects requires supporting:

PERSONAL, INSTEAD OF GENERIC REPRESENTATIONS OF VISUAL PROPERTIES.

Designers should be able to use personal samples to create representations of visual properties. For example, they can create personal color representations based on collected images, or personal palettes. They can also start from previous artifacts, instead of generic templates, to create a new layout. Personal representations hold meaning relevant for the user, and for the current composition.

LINKS AMONG VISUAL PROPERTIES

Changing values of visual properties depends on other visual properties in the composition. Designers establish links among properties. They manipulate these links throughout their composition. They know and articulate these links, but execute them manually in current visual composition tools. They cannot embed these links as part of their composition. Instead, they have to change values of individual properties separately and repeatedly in the composition.

RULES FOR EVOLVING VISUAL PROPERTIES.

As they create their compositions, designers define fixed, constant visual properties. They define other properties as flexible, and change their values in space, for several objects in the composition. They can also change their values in time, across related compositions. Designers establish rules to define how these properties change in these situations. In current tools, designers define these rules verbally, but cannot embed them in the tool. They need to change the property value for each instance manually.

INTERACTIONS WITH VISUAL PROPERTIES IN THE COMPOSITION, NOT IN SEPARATE WINDOWS OR DIALOG BOXES.

Designers need to maintain a constant conversation with the content as they create their composition. For this reason, manipulating visual properties depends on the content, and on the current state of the document. Designers thus need to manipulate properties in the document. They need to experiment with several values of these properties, before getting their final result.

In the following, I present factors that affect how designers create their visual compositions. First, perception helps identify how we sense and comprehend the components of a visual stimulus. Then, design guidelines determine best visual composition practices. Finally, existing representations of visual properties such as color, layout and typography affect how designers use these visual properties in their compositions.

2.3 PERCEIVING VISUAL PROPERTIES

Our perception of visual properties affects how we interpret, internalize and manipulate them in creating compositions. In perceptual processing, visual properties are component parts of the image we perceive. Gestalt psychologists rely on phenomenological evidence to support that perception follows a holistic approach. The properties of the whole object are different from the properties of its component parts. Also, detecting the whole object facilitates and precedes the detection of its components.

Recent studies (Wolfe and Robertson 2012) showed early perceptual processing of object properties. Two observations supported this: specialized populations of neurons tuned to specific colors, disparities, orientations, or spatial frequencies. Also, selective attention to a specific property was found efficient in sorting and classification tasks. In these observations, there was no significant interference from properties irrelevant to the task.

2.3.1 *Perceptual Organization*

Perceptual organization is *“the process by which particular relationships among potentially separate stimulus elements are perceived and in turn guide the interpretation of those elements”* (Pomerantz and Kubovy 1986). Three major approaches to perceptual organization suggest different roles for properties as we perceive a visual stimulus. All three approaches agree on the early processing of these properties as we sense and comprehend visual objects.

THE STRUCTURALISTIC APPROACH:

The perceived whole is equal to the sum of its part. For this reason, perceptual organization and perception are separate. Different sensory organs perceive individual components. The resulting sensations have a trace: memory images. These traces are evoked when the same sensory object is encountered again. The organization then is done separately when the brain creates links between the collected memory images (Pomerantz and Kubovy 1986). Thus, each visual property has a memory image that determines how it is perceived. Interaction between these properties depends on the existing links between memory images. From a structuralistic point of view, different observers will perceive different links between visual properties depending on their memory images.

THE HELMHOLTZEAN APPROACH:

The Helmholtzeans and the Structuralists provide similar definitions of the roles of sensation and memory images. The Helmholtzean approach also acknowledges sensation to be the first step in perception. Memory images acquire associations that help recognize what we see. The Helmholtzean approach is more flexible. The organization of sensations creates a whole that is larger than the sum of its perceived components. This approach adds the likelihood principle: *“sensory elements will be organized into the most probable object or event in the environment consistent with the sensory data”* (Pomerantz and Kubovy 1986). Thus, visual properties in the Helmholtzean approach, like in the structuralistic approach, are sensations that trigger specific associa-

tions. They are associated to their closest resemblance based on the sensed data.

THE GESTALT APPROACH:

Designers use Gestalt principles to guide the manipulation of visual elements (section 2.5.3). The Gestalt approach states that *“the whole is more than the sum of its parts”*.

Visual properties, as components of the whole stimulus, are perceived holistically rather than separately. The notion of holism led to new theories that explain the nuances of considering the whole larger than its component parts. Emergent features (Fig. 5) are visual properties that emerge when different parts combine into a whole. For example, a set of trees closely spaced is visually grouped into a forest. This forest has new visual properties such as density, that individual trees do not have (Wagemans et al. 2012). Other examples of emerging features include symmetry, closure and regularity. These emergent features, also called holistic properties, exist along with properties of individual components (Wagemans et al. 2012). The Gestalt approach does not ignore the importance of visual features that belong to individual components. It suggests a new type of visual properties that emerge from a specific organization of the components.

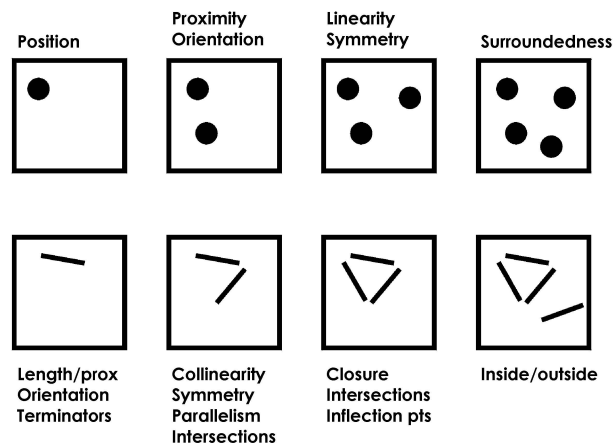


Figure 5: Top row shows how novel features emerge as additional dots are added to a stimulus, while the bottom row shows the same for line segments.(Pomerantz and Cragin 2015)

These three approaches to perceptual organization provide different insights about the perception of visual properties. They base their definitions on the structure and function of the brain, and on the role of learning versus innate functions in perception. The Gestalt approach highlights the importance of the stimulus' structure in defining what we perceive. Thus, a new type of visual properties, emergent features, are the result of a specific organization of the stimulus ele-

ments.

Another approach to perceptual organization is direct perception (J. J. Gibson 1950, 1966, 1979). In direct perception, the organization does not happen in the brain. We perceive the organization the stimulus already has. Perceptual organization does not need more information from memory to interpret the stimulus (Pomerantz and Kubovy 1986). Perceptual learning further explains how the perception of visual properties evolves over time.

2.3.2 *Affordances and Perceptual learning*

Gibson's ecological view of perception is based on the theory of affordances. He focused on the "*interactive relationships of behaving agents with systems in their environments*" (Greeno 1994). Gibson refers to visual properties as: "object qualities". He explains the particularity of the ecological view:

"Orthodox psychology asserts we perceive these objects insofar as we discriminate their properties or qualities [...] But I now suggest that what we perceive when we look at these objects are their affordances, not their qualities (Gibson 1979)."

In Gibson's view, we do not perceive visual properties such as size and distance. We perceive instead if the object is within arm's reach, or whether it can fit into our grasp (Adolph and Kretch 2015).

Affordances are the invariant combination of perceptible properties that compose the object. Gibson, like Gestalt psychologists, also claims it is easier to perceive the whole invariant unit first, compared to perceiving all the variables separately. Gibson's affordance theory further explains why meaningful perception is direct by considering the "*ecological object*" of perception to be value-rich (Gibson 1979). The object's affordances are a set of properties taken with reference to the observer. These affordances are preconditions for activity. Their presence does not mean the activity will occur, but contributes to the possibility of its occurrence (Greeno 1994). Affordances are always there to be perceived, but the observer, depending on her needs, may or may not perceive them (Gibson 1979).

Eleanor Gibson further investigates these changes. She introduces perceptual development: the process of learning about and discovering new affordances as action capabilities change (Adolph and Kretch 2015). "*Distinctive features*" in perceptual learning are a type of visual properties. These are: "*the relational contrasts or minimal set of attributes that distinguish one kind of thing from another*" (Adolph and Kretch 2015). These properties are invariant under certain transformations. Thus, all distinctive features are visual properties of the perceived ob-

ject, but not all visual properties are distinctive features.

Perceptual learning is a process of “*differentiating previously vague impressions*” (Adolph and Kretch 2015). The goal of perceptual learning is to increase the specificity of the perceived information through differentiation. For example, a novice color practitioner can only distinguish primary and secondary color hues. An experienced color practitioner perceives differences that correspond to slightly different shades of the same hue, to the amount of cyan in a color, or to the result of using the same color on different media. With practice, the color practitioner learns to distinguish targets of perception she could not discriminate before.

Perceptual organization suggests that visual properties are component parts of the stimulus we perceive. It explains how we perceive and comprehend different types of visual properties. In particular, the concept of “*emergent features*” differentiates two types of visual properties. The first type includes visual properties inherent to the stimulus or to its components. The second type includes visual properties that are results of the current organization.

The theory of affordances suggests that we perceive how we can manipulate visual properties depending on our current action capabilities. Perceptual learning introduces the term “*distinctive features*” to describe the object properties an observer can differentiate. The specificity of these properties increases with experience.

Perception and the observer’s knowledge of how to manipulate visual properties affect what they perceive, and how they manipulate these properties. Perceptual organization highlights the role of visual properties in comprehending what we see. Affordances and perceptual learning link what the observer knows to the number of distinctive features they perceive within a stimulus.

In this thesis, I focus on design practitioners who have knowledge and experience in creating visual compositions. Design guidelines affect the choice of visual properties designers use to create their compositions. The representations and tools the designer has at hand also determine how they can manipulate these properties, and how they can use them in their compositions.

2.4 VISUAL COMPOSITION GUIDELINES

“Rules can be broken –but never ignored”

David Jury (Samara 2007)

We want to identify how design guidelines affect the composition process. Designers create compositions around a visual concept. Samara describes the work of a graphic designer as being around communicating a clear message. The communication tools a designer has at hand include: *“imagery, symbols, type, color, and material -whether it’s concrete, like printing on a page, or somewhat intangible, like pixels on a computer screen or light in a video.”* (Samara 2007).

Lauer and Pentak (Lauer and Pentak 2012) describe four design principles that guide designers as they create visual compositions. For each principle, dedicated guidelines indicate the visual properties the designer can use to create specific effects.

UNITY:

The designer should provide some type of organization to relate various elements. It is the creator’s task to ensure these elements belong together, that the whole is larger than the sum of its individual components (Fig. 6). Visual properties the designer manipulates to achieve unity include repetition, alignment, continuation and proximity (Fig. 10).



Figure 6: Monochromatic picture uses color to reinforce unity. Damon Winter. Personal photograph from Iceland. Communication Arts, May/June 2005. (Lauer and Pentak 2012)

VARIETY:

While unity creates coherence within the design, variety complements it and brings interest to the composition by changing the character of some design elements (Fig. 7). The designer manipulates visual properties such as size, thickness, value, color, angle, orientation and texture to create variety in the composition (Fig. 10).

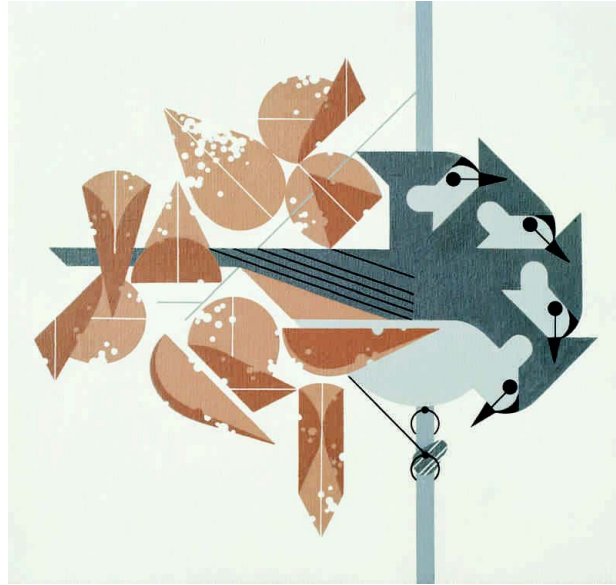


Figure 7: Geometric unity with variations on a theme. Charley Harper. Original painting for Titmouse Tidbit (Tufted Titmouse). 1975. Acrylic on canvas, 1' 1" × 1' 2". (Lauer and Pentak 2012)

EMPHASIS:

Attracting the viewer's attention to what is most important requires creating a focal point. This point of emphasis will also encourage the viewer to look at the details of the design (Fig. 8). Visual properties such as contrast and position help the designer define a focal point and achieve emphasis in the composition (Fig. 10).

BALANCE:

This principle refers to the distribution of visual weight in a composition (Fig. 9). The visual properties of repetition and symmetry contribute to creating balance in a visual composition (Fig. 10).

Guidelines encourage designers to manipulate specific visual properties to create unity, variety, emphasis and balance in their compositions. These guidelines are more often examples and suggestions, rather than unbreakable rules. Their role is to suggest starting points for designers to communicate their personal visual concepts.

Designers take these guidelines into consideration as they create their compositions. More specific guidelines propose tools and representations for particular visual properties. I focus on three visual properties: color, typography, and layout.



Figure 8: All aspects of compositions attract attention to the grapefruit as a focal point. Susan Jane Walp. Grapefruit with Black Ribbons. 2000. Oil on linen, 8" × 8¼". Tibor de Nagy Gallery, New York. (Lauer and Pentak 2012)

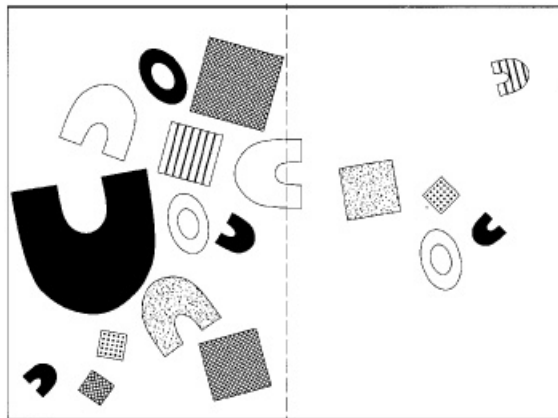


Figure 9: An unbalanced design creates uneasiness for the viewer. (Lauer and Pentak 2012)

Unity	Repetition	Alignment	Proximity
	Continuation		
Variety	Texture	Thickness	Orientation
	Angle	Color	Size
	Value		
Emphasis	Contrast	Position	
Balance	Repetition	Symmetry	

Figure 10: Design guidelines suggest changing specific properties to apply the principles of unity, variety, emphasis and balance.

2.5 REPRESENTING VISUAL PROPERTIES

Designers create and use tools and representations to manipulate color, typography and layout. Some of these representations, such

as the color wheel, transferred to graphical user interfaces. They thus affect how users manipulate these visual properties as they create online compositions.

2.5.1 Color

“If one says “ red ” and there are fifty people listening, it can be expected that there will be fifty reds in their minds. And ... all these reds will be very different. Colors present themselves in continuous flux, constantly related to changing neighbors and changing conditions.”

Joset Albers (Albers 1963)

Color carries psychological and emotional meanings that vary across recipients. Color affects the legibility of type, hierarchy, and how viewers establish connections between visual elements (Samara 2007). For this reason, the choice of color strongly affects the concept a designer wishes to communicate. Skillful color manipulation requires comprehending the physical, perceptual and cognitive characteristics of color.

2.5.1.1 Understanding Color

Color is a singular visual phenomenon. Understanding color theory requires collecting fragments of theories from various disciplines. These different points of view should be consistent within themselves, with each other, and with what we see. Physics explains the cause of color. The literature of art education informs us about the beauty and harmoniousness of color. From biology, we learn about the functional roles of color in nature. Social sciences study color names and the relationship between color names and our experience of seeing color (Sloane 1989).

Figure 11 is a drawing by Newton of his experiments with light. The two main components of the apparatus are in the center of the illustration: the prism and the lens. Sunlight passing through the prism creates an image of a rainbow on the wall. In the upper left of the sketch, we can see points Newton drilled in the wall to allow parts of the rainbow to continue through to a second prism. The image produced on the second surface was not another rainbow, but remained in the same color passed from the first rainbow. Newton concluded that passing through the first prism separated sunlight into its fundamental components: no other changes happened when passing through the second prism (Wandell 1995).

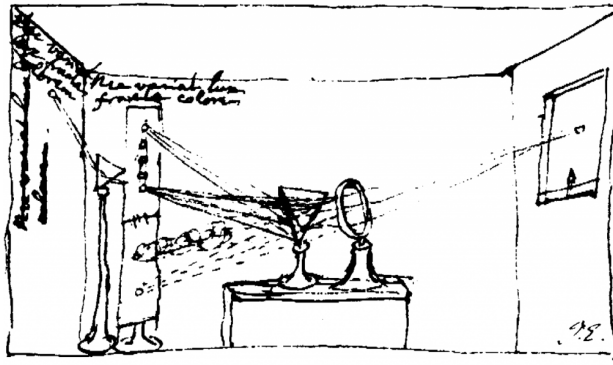


Figure 11: Newton's drawing of his Prism Experiment (Wandell 1995)

Newton's experimentations presented color as a property of light. Subsequent hypotheses in physics about what causes color defined it: "in terms of light, and light in terms of waves. The waves were identified as an electromagnetic phenomenon." (Sloane 1989)

Physics provides insights about the cause of color. Understanding the nature of color requires taking sensation and perception into account. Russel (Russel 1948) illustrates this importance in the following example:

"Until the nineteenth century people were uttering meaningless noises when they said that blood is red, because nothing was known of the correlation of wave lengths with sensations of color. This is absurd. It is obvious that red has a meaning independent of physics, and that this meaning is relevant in collecting data for the physical theory of colors, just as the pre-scientific meaning of "hot " is relevant in establishing the physical theory of heat."

Russel argues here that explaining "red" with a wavelength is not complete. It is not possible to define a color objectively, regardless of how the eye experiences and encounters this color. Color vision occurs when photoreceptors convert light into nerve signals. These photoreceptors, contain photopigments responsible of detecting specific color wavelengths. Color perception requires cross-referencing nerve signals coming from multiple types of photoreceptors (King 2005).

Goethe (Goethe 1970) focused on color as a phenomenon he studied through a series of practical color mixing experiments. These results demonstrate how the appearance of color to our eyes, and our brain's interpretation of what we perceive affect our moods and emotions following this experience.

Further studies demonstrated that characteristics such as cultural background, affect our interpretation of color. Kubat et al. asked 23 english-speaking participants to choose one of two color terms to describe an ambiguous color. Participants' perception of ambiguous colors shifted towards typical colors of common objects of a similar color (Kubat, Mirman, and Roy 2009). In this experiment, participants' semantic knowledge of typical colors of common objects affected their description of the color they saw.

Our personal experiences affect the color we perceive, and the color we perceive affects our moods and emotions. Montazeri et al. found that color affects the salience of an object, which for their experiment, affected participants' recycling behavior: 88% of participants presented with a green recycle bin recycled, while only 52% of participants presented with a gray recycle bin used it (Montazeri et al. 2012).

In visual arts, color theory is based on elements from physics, perception, and cognition. It is a body of prescriptive rules that guide color manipulation. The technical aspect of color theory in visual arts predicts the results of specific color mixtures. Its aesthetic aspect ensures harmoniousness of color arrangements (Sloane 1989).

From a technical perspective, color theory starts from the assumption of three primary colors. Secondary colors result from mixing two different primary colors. Tertiary colors result from mixing all primary colors. If we assume that primary colors are blue, yellow and red, secondaries are orange (red and yellow), green (yellow and blue), and purple (blue and red). The tertiary colors result either from directly mixing all three primary colors, or from mixing two secondary colors. The range of tertiary colors includes brown, russets, olives, grayish browns, and bluish grays (Sloane 1989). Figure 12 is a representation of primary and secondary colors Itten used to teach novice artists about color theory (Itten 1970). Artists often assume red, blue and yellow as primary colors. This assumption is based on mixing pigments and is often called subtractive mixing. Thus, mixing all colors forms black. Additive color mixing is based on color as light waves, and mixing all three primary colors forms white. In additive mixing, the three primary colors : red, blue and green.

From an aesthetic perspective, color theory provides principles and guidelines to obtain harmonious colors. Goethe (Goethe 1970) introduced the completeness principle: *"the eye seeks for a colorless space next every hue in order to produce the complementary hue upon it"*. By this principle, when the eye perceives a color, it produces another color. This new color along with the first comprehend the whole color

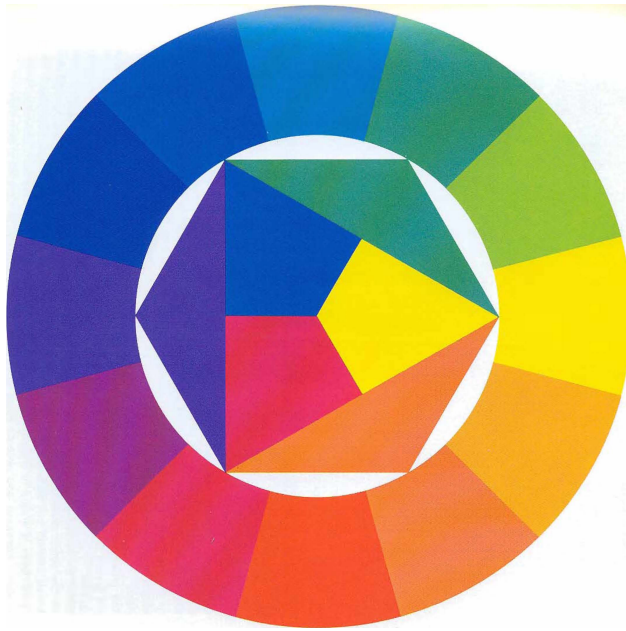


Figure 12: The 12 hues color cycle (p.31)(Itten 1970)

scale. (Goethe 1970)

The principle of completeness identifies complementary colors. Goethe uses the colorific circle (Fig. 13) to identify complementary colors. Thus, by moving a diametric index around the circle, its two ends indicate complementary colors. Contrasting complementary colors is “harmonious” because it carries the conditions for completeness (Goethe 1970). Complementary colors summarized in the following three contrasts:

- Yellow and Red-Blue (Purple)
- Blue and Red-yellow (Orange)
- Red and Green

Goethe also classifies colors in one of two sides, using the expressions *plus* and *minus*. Warm colors (red to orange to yellow) are on the *plus* side, and cool colors (green to blue to purple) are on the *minus* side (Goethe 1970).

Color theory is relevant in visual arts because of its practical implications. The quality of a color representation depends on its practical usefulness when mixing and arranging colors.

2.5.1.2 Color Representations

Color Representations serve as visual tools to communicate color theory principles. The color wheel is one of the most popular representa-

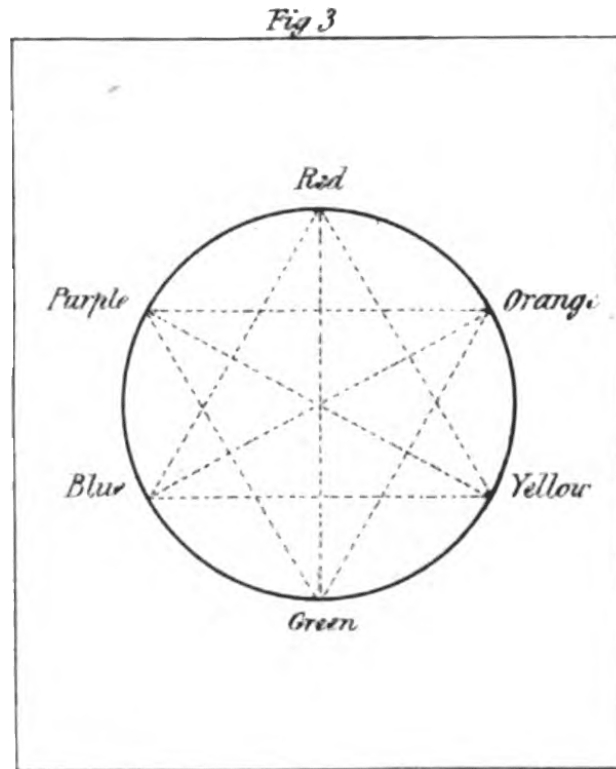


Figure 13: Goethe's Colorific Circle (Plate.1 Figure.3)(Goethe 1970)

tions of color. Newton created one of the first color wheels (Fig. 14) by placing purple between red and blue. This created a two dimensional color representation from the spectral one dimensional representation of rainbow colors (Baumann and Verlag 1994).

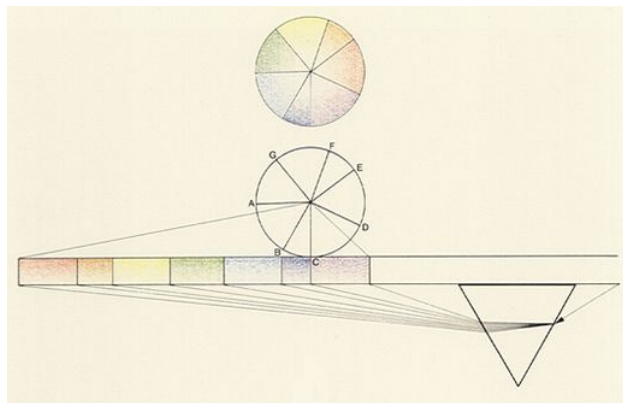


Figure 14: Newton's color circle

The color wheel communicates and summarizes color relationships (Fig. 13). The color wheel is also a tool novice artists create when learning these principles. Itten suggests an exercise where his students create the

12 parts color wheel (Fig. 12). He describes this representation as a tool to visualize the 12 hues, any other tones in between, and the relationships between these hues: *“The twelve hues evenly spaced, with complementary colors diametrically opposite each other. One can accurately visualize any of these twelve hues at any time, and any intermediate tones easily located.”* (Itten 1970)

Albers suggests another representation of primary, secondary, and tertiary colors: the equilateral triangle (Fig. 15).

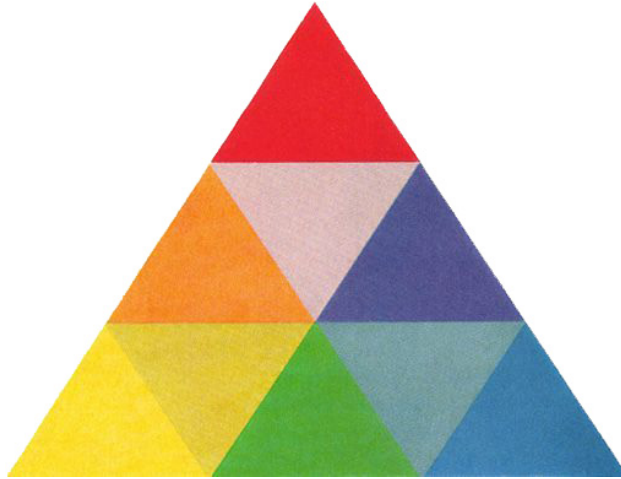
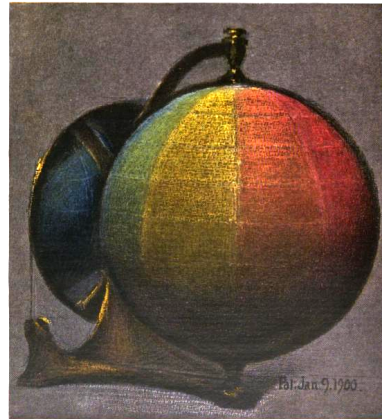


Figure 15: The Equilateral Triangle p.191 (Albers 1963)

The three primary colors are on the edges. They have the highest contrast and thus are the most separated colors in the triangle. The three secondaries are in the middle of the outer edges. The three tertiaries present the least contrast and are in the center. Albers suggests starting with this organization of the equilateral triangle. He encourages his students to create and compare different organizations (Albers 1963).

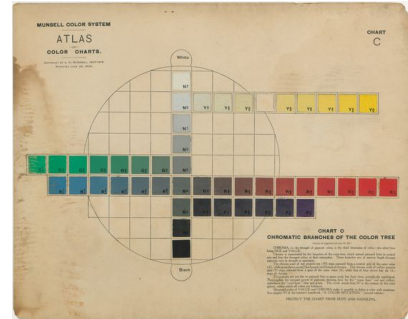
The Munsell *Sphere* (Fig. 16.a) is a three-dimensional representation that visually separates hue, value, and chroma (Munsell 1905). Munsell introduced these three qualities of color. He gave the example of a faded red cap to further explain their practical meaning. The hue represents the redness of the cap. Hue is the quality by which we can distinguish one color from another. In this case, it is what distinguishes a red cap from a blue cap. The purity of this red is the chroma. This quality reflects the strength of the color. In this example, the cap's red color has lost in chroma and is getting closer to a neutral gray. The amount of red in the cap is the value. This quality differentiates a light color from a dark one. In this example, it makes the difference between a light red and a dark red cap. Each of these

three elements contributes to the red we see (Munsell 1905).



A BALANCED COLOR SPHERE

(a) The initial balanced Color Sphere (Munsell 1905)



(b) Chromatic branches of the Color Tree (Munsell 1915)

Figure 16: Munsell Color Representations

Later, Munsell proposed a different representation: the *color tree* (Fig. 16.b). The initial sphere is unbalanced to account for perceptual distances between hues (Munsell 1929). Tree branches surrounding the sphere represent chromatic branches for each hue.

The Munsell *color tree* is a visual representation of the three color dimensions: hue, chroma, and value. It also visualizes the difference in distance between chromas of different hues: “HUES of stronger CHROMA extend some steps out beyond the surface, like immense buildings on a very small Earth; or like level limbs on a Color Tree extending through and beyond the surface of a sphere within which the tree has grown” (Munsell 1929).

Ultimately, the *color tree* became a physical tool for color professionals to identify and create colors (Edward R. Landa and Mark D. Fairchild 2005). For example, the Munsell tree is heavily used in soil-profile characterization (Fig. 17).

Other color representations address specific display media. RGB (red, green, blue) and CMY(cyan, magenta, yellow) two popular models for color production on digital and print media. They are complementary models: the primary colors of one are the secondary colors of the other (Fig. 18). RGB is based on additive mixing, the choice of colors here is also linked to the types of cones in the human eye, and to color as light waves (Fig. 18). CMY is based on subtractive color mixing, and on color as pigments (Fig. 18) (Adobe 2000).

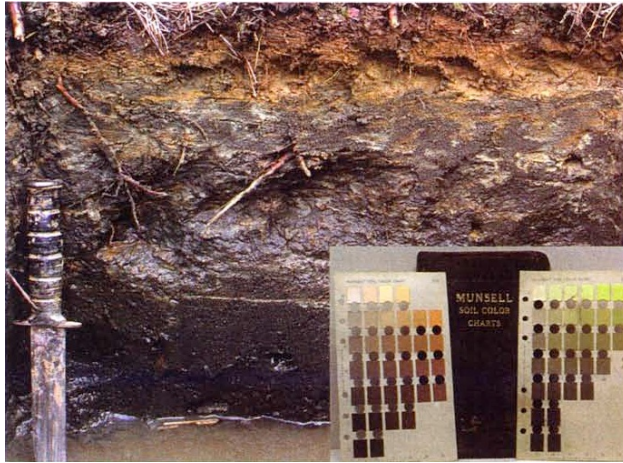


Figure 17: Munsell Soil Color Charts (Edward R. Landa and Mark D. Fairchild 2005)

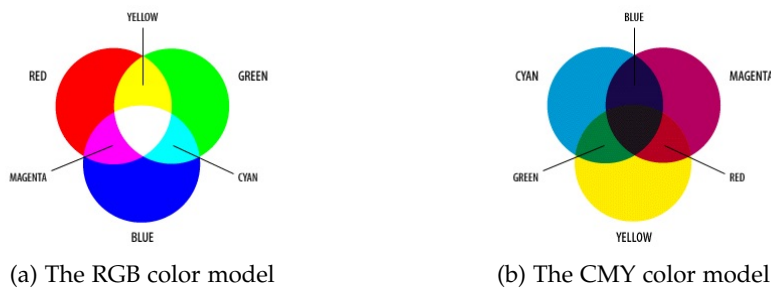


Figure 18: RGB and CMY color models

CMY is used for printing colors. In practice, it is impossible to produce a pure black by mixing cyan, magenta and yellow in equal proportions. For this reason, printers add black ink to boost grays and shadows. Thus, the CMY model is replaced in practice with the CMYK process.

RGB and CMY cannot reproduce all the colors we can see (Fig. 19). Several RGB colors cannot be recreated in CMY(K), and some CMY(K) colors cannot be reproduced in RGB. The gamut is the range of all colors a model produces. For the RGB model, the gamut depends on the device. This dependence can create problems when representing the same colors across several devices (Adobe 2000).

More recent color models reproduce the entire gamut of colors we can see. The CIE color model (CIE stands for comité internationale de l'éclairage) is device independent and covers the whole range of colors we can see (Fig. 19).

The Munsell color system affected color representations on digital displays. The HSB/HLS color model (hue, saturation, luminosity/brightness) is based on the Munsell system's three color dimen-

sions: hue, chroma, and value. Hues range from red to green, and from 0 to 360 degrees on a circular pattern. Saturation in this model is equivalent to chroma in the Munsell system and ranges from 0 to 100% on a linear scale. Luminosity (or brightness) also ranges from 0 to 100 % on a linear scale and represents value in the Munsell color system (Adobe 2000).

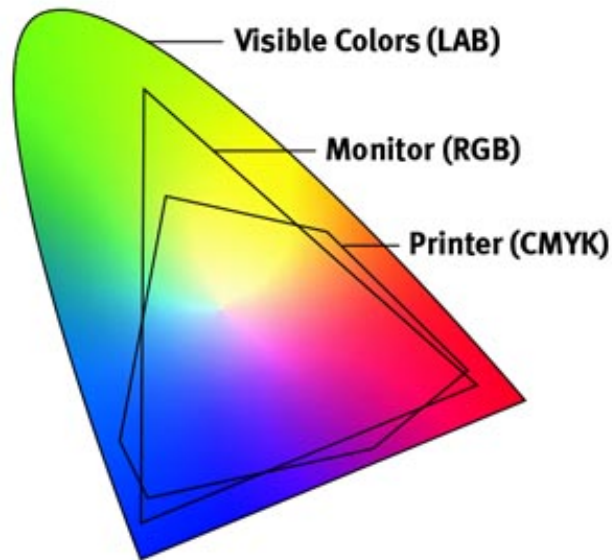


Figure 19: The RGB and CMYK gamuts (OKI Data 2016)

Color models for digital color representations changed from adapting to individual media, to adapting to human color perception. This enhanced the persistence of color when changing monitors. Yet, conversion problems still exist when moving from a color model to another.

Digital color representations differ from color representations in visual arts. The goal of digital color models is to optimize the number of perceivable colors they reproduce on a display. Yet, they provide little guidance on how to use these models to mix, arrange, and convert colors from a model to another. We do not have visual color representations that predict what mixing red and blue would mean in the RGB model, and how harmonious the result would be if arranged with a mixture of red and green.

Color representations based on the HSV model use the color wheel, with two linear sliders for saturation and luminosity. The color wheel communicates information about primary and secondary colors. The whole representation, the color wheel and the two sliders, is less useful than the Munsell color tree for example. The Munsell tree connects all three qualities. It also includes chromatic distances for the same

hue and shows the distance between different hues.

Digital color models can now reproduce all perceivable colors. Yet, we still lack informative representations of their technical or aesthetic implications. We do not have color representations that inform and guide the manipulation of color using digital color models such as RGB or CMY.

2.5.1.3 *Using Color in visual compositions*

In visual arts, color principles and visual representations should provide guidance for color manipulation. These rules help predict and orient color experimentation. Color principles such as complementary, warm and cold colors specify harmonious color arrangements.

Artists and art teachers highlight the importance of personal experimentation with color. Joseph Albers taught laboratory style courses where students understand color through practical exercises. His goal was the development of his students' sensitive eye for color. The eye and the mind need to be prepared, through productive exercises, to recognize beautiful color orders (Albers 1963).

Albers introduced a color manipulation principle in several of his color experiments: the relativity of color. He called color: "*the most relative medium in art*" (Albers 1963). To demonstrate this principle, Albers suggested an exercise where he made the same color look different. He placed two small rectangles of the same size and of the same color on large rectangles of two different colors (Fig. 20). This setup differentiates the influenced color (the color of the small rectangles) and the influencing color (the background color) (Albers 1963). Understanding how colors affect each other helps designers manipulate color relationships to create desired effects.

Color systems, theories, and representations are tools color practitioners use to build a personal color experience. Gaining this experience permits recognizing and creating aesthetically harmonious color orders. Itten (Itten 1970) identifies three directions for gaining knowledge about the aesthetic aspect of color effects.

- Impression (visually):
This consists of observing color in nature and reproducing the finest details of these colors. Itten gives the example of impressionists, such as Monet and Renoir. They were interested in reproducing the color vibrations light produces on surfaces at different times of the day (Itten 1973).

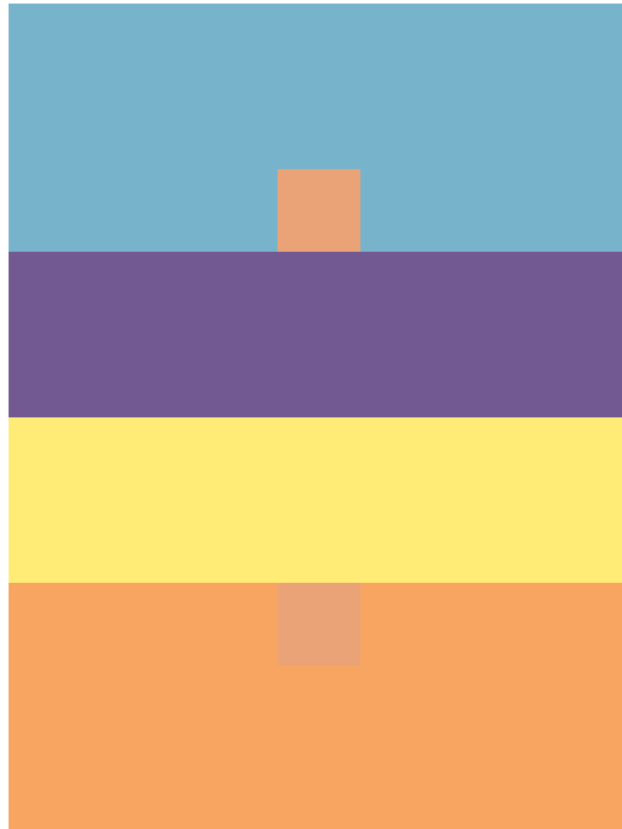


Figure 20: The relativity of color (p.76)(Albers 1963)

- Expression (emotionally):
Color acts here as a tool for expressing the artist's emotions.
- Construction (symbolically):
The symbolic meanings of color make it an important medium to convey the artist's ideas.

These three directions are interrelated. Itten says: *"Symbolism without visual accuracy and without emotional force would be mere anemic formalism; visually impressive effect without symbolic verity and emotional power would be banal imitative naturalism; emotional effect without constructive symbolic content of visual strength would be limited to the plane sentimental expression"* (Itten 1970). The impressive, expressive and constructive effects of color complement each other. They allow each artist to build and convey a unique color manipulation experience.

When color practitioners internalize color rules, they create their own color representations. Kandinsky, a prominent painter, created famous color compositions with bold shapes and primary colors (Fig. 21).

Like color theory, color representations guide color manipulation, but do not define the final color choices. Designers use these represen-



Figure 21: Kandinsky's concentric circles

tations to classify colors (Fig. 16), and to predict the results of mixing specific colors (Fig. 20).

2.5.2 *Typography*

Typography is the process through which language becomes visible. A typographic style is *“the power to move freely through the whole domain of typography, and to function at every step in a way that is graceful and vital instead of binal”* (Bringinghurst 2002). Typographers learn the fundamentals and principles of using and designing typefaces. They have the power to choose and create typefaces according to the goals they set for themselves.

Working with type requires understanding the fundamental characteristics of typographic systems. For example, western alphabet letters belong to a system with specific visual relationships. At a standard reading size, the eye perceives letters to be all the same weight, height, and width. This uniform style discourages distraction while reading (Samara 2007). Two fundamentals of typography that apply to digital environments are:

TYPOGRAPHY FORMS HAVE A DUAL NATURE:

Typography has a semantic and an aesthetic nature (Cullen 2012). On a formal level, typography forms function like lines, shapes, and texture. They affect the visual impact of the composition. On the content level, typography forms communicate a verbal message (Evans and Thomas 2012). Composing with typography requires considering these two dimensions while representing and manipulating word forms.

LEGIBILITY AND EXPRESSION:

The trade-off between legible and expressive type can be seen through two main movements: functionalism and expressionism. Functionalism measures the appropriateness of a type by its ability to communicate information to a specific audience. Expressionism gives more attention to experimental, expressive and ornamental possibilities (Carter et al. 2015). Bringhurst suggests that “*typography must often draw attention to itself before it will be read. Yet in order to be read, it must relinquish the attention it has drawn*” (Bringhurst 2002). The designer should balance expressivity of a type form and its legibility.

Typography principles provide prescriptive rules to guide typographic choices. We focus on principles that define the impact of type visual properties.

TEXT ELEMENTS WITH THE SAME TREATMENT ARE ASSUMED TO BE RELATED:

Typographic properties define the hierarchy of informational components. Typographic blocks that have the same choice of typeface properties are perceived to mean similar things (Samara 2007). Content predetermines logical relationships of emphasis and value between parts of text. Typography should clearly express these relationships (Tschichold, Kinross, and Hendel 2006).

TYPE IS AFFECTED BY HOW IT IS READ:

Typography decisions depend on how the text will be read. It is important to consider cases where reading does not follow the default reading directions of the language. For example, the typographer should not always assume that western alphabet will be read starting top left. Another plausible order might start from headings, possibly to the remaining text, in order of importance (Tschichold, Kinross, and Hendel 2006). The rhythm of strokes and spaces of a typeface affects its optimal spacing. It then affects the optimal spacing between words, lines, and paragraphs. Arriving at this optimal spacing creates a “*consistent gray value in text*” (Samara 2007).

TYPE IS A COLLECTION OF LINES, DOTS AND SHAPES:

Type behaves like its simplified components. The dot defines a location in space. It can be used for alignment or to activate spaces in a composition. The line can be used for separation, enclosing or emphasis. The shape can operate as inclusions or details among letters or words. The shape can support clusters of text, by acting like a field upon which type lies (Samara 2007).

2.5.2.1 Digital Typography: Representation and Manipulation

Digital typography tools introduced and encouraged new representations of type. The screen introduced new aesthetic and technical considerations. Issues related to the digital medium incited typographers to challenge some of the initial rules of legibility. They developed hybridized forms and used computer-based and algorithmically-driven typography (Staples 2000).

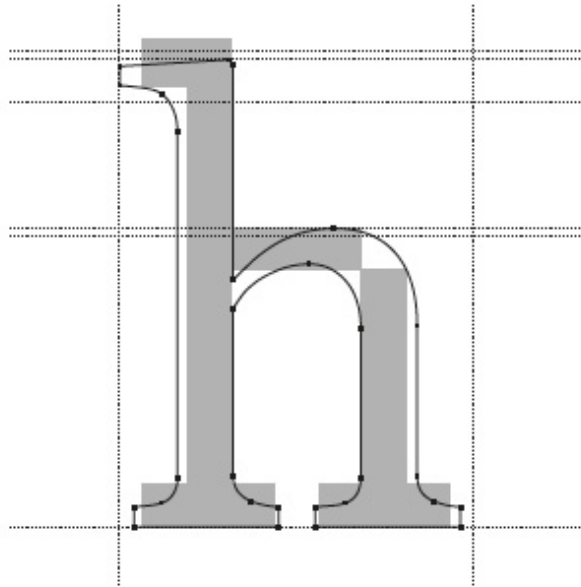


Figure 22: Text size Georgia h as a bitmap and as an outline. (Carter et al. 2015)

Rendering type on the screen requires going from an outline letterform stored as a Bézier curve, to a rasterized -pixelized- version (Fig. 22). Techniques such as anti-aliasing and hinting, improve the rendering of fonts on the screen. Antialiasing creates a smooth curvature illusion to replace the stair-step edges created by pixels. Hinting creates the best possible image of type at different point sizes. Hinting selectively activates specific pixels at different point sizes to improve type legibility on low-resolution devices (Carter et al. 2015).

2.5.3 Layout and Visual Organization

“To design is much more than simply to assemble, to order, or even to edit; it is to add value and meaning, to illuminate, to simplify, to clarify, to modify, to dignify, to dramatize, to persuade, and perhaps even to amuse.”

Paul Rand (Cullen 2007)

The designer creates a unique concept that communicates a clear and concise message. For this purpose, the white space on paper

or on a digital document is a canvas to create meaning. Designers build an order system for visual elements. They assign a level of importance and activity to color, form, image, space, and typography (Cullen 2007). Layout is the conversation that happens between different visual elements. Designers define rules for how visual elements interact in space. Each designer finds a unique concept and defines how arranging visual elements can best communicate it. Cullen identifies two fundamental elements to guide designers through this process (Cullen 2007):

HIERARCHY:

The designer creates an ordering system of visual elements the viewer can recognize. This ordering system evolves around a focal point that first attracts attention. Starting from the focal point, other visual points are ranked by importance. The designer also identifies visual elements that belong to the same group. In this step, the designer builds consistency by defining relationships between visual elements.

COMPOSITION:

The designer manipulates composition factors: properties such as position, tension, repetition, and scale. The designer selects and manipulates composition factors to create the design hierarchy. Composition factors are visual properties the designer uses to create a visual concept for organizing content.

Contrast distinguishes the visual elements that interact in the compositional space. Contrast makes the ordering system apparent to the viewer.

Hierarchy and composition help the designer build an ordering system the viewer can decrypt and interpret. The designer manipulates composition factors to order visual elements in space.

Principles based on perception and aesthetics help the designer build a hierarchy where composition factors activate visual elements in the composition space. These principles maintain the link between the viewer's perception, composition factors, visual elements and the designer's intention. It is important for designers to be aware of these principles. Breaking them is allowed if the composition concept justifies it.

The principles of Gestalt relate to "*the way things come together*". These principles state that the interaction of visual elements has a higher dynamicity than each element individually (Evans and Thomas 2012). The following Gestalt principles guide the manipulation of composition factors for visual organization:

HEAVY VISUAL ELEMENTS ATTRACT ATTENTION FIRST:

This principle depends on a composition factor: visual weight. Heavier elements will attract the viewer's attention sooner and for longer periods.

Visual weight depends on the visual element and on other composition factors. Position affects visual weight. Visual elements near the central horizontal or vertical center of the composition have more visual weight.

Depth also affects visual weight: *"the greater the depth of the visual field, the greater the weight it carries"* (Arnheim 1974). Visual weight also depends on size: larger objects are heavier.

SIMILARITY IS A PREREQUISITE FOR NOTICING DIFFERENCES:

Arnheim suggests that comparisons, separations and connections between visual elements happen only when the context provides sufficient background (Arnheim 1974). This principle also links to composition factors that predict prominent visual groupings. Size, shape, color and position all contribute in creating detectable groupings of visual elements.

ORIENTATION DEPENDS ON THE FRAME OF REFERENCE:

This principle helps the designer answer the following question: *"What conditions must visual form meet for an image to be recognizable?"* (Arnheim 1974)

Orientation defines how the designer expects the viewer to relate visually to the image. A default frame of reference consists of *"perceiving the world standing or sitting perpendicular to the ground"* (Evans and Thomas 2012).

Changing the orientation of a triangle or a rectangle without changing the frame of reference does not change the shape (Fig 23). It is perceived as deviated from its more *"normal"* position. Changing the orientation of a square by the same angle changes the square into a different shape (Arnheim 1974).

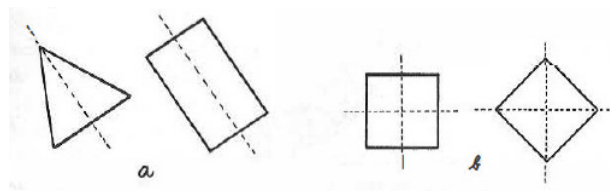


Figure 23: Changing orientation depends on the frame of reference. (Arnheim 1974)

In this section, I point at composition factors, spatial properties such as size, position, orientation and layering. Layout guidelines emphasize the role of this type of properties for visual organization.

Layout guidelines help predict the results of certain arrangements of composition factors. These guidelines do not replace the importance of experimenting with composition factors to build an ordering that conveys the designer's concept. Experimentation requires organization tools, such as the grid, that keep composition factors visible throughout the organization process.

2.5.3.1 *The grid: a tool for representing visual organization*

The grid is a tool to visually represent composition factors. The grid keeps these factors visible when defining how visual elements interact in the composition space. This visual representation is often removed from the final composition (Fig. 24).

The grid is a network of horizontal and vertical intersecting lines that transform the page into a framework of areas (Lauer and Pentak 2012).

In the 20th century, the grid shifted from focusing on external appear-

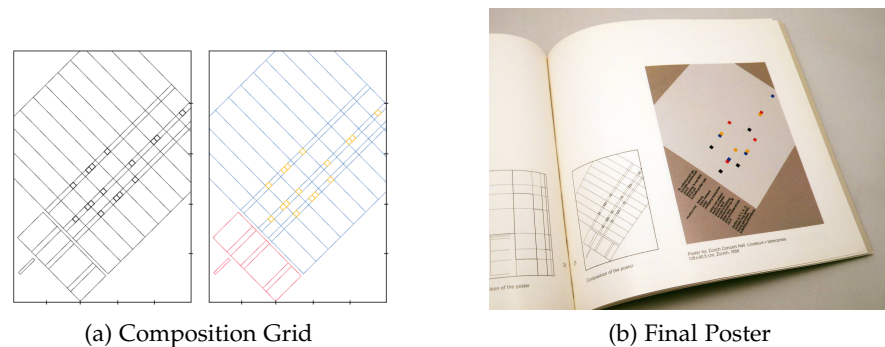


Figure 24: Zürich Tonhalle Musica Viva Concert Poster, 1958, Joseph Müller-Brockmann

ance to focus on structure. It became a tool for representing the process of structuring and delivering rational information (Williamson 1986).

Bauhaus designers presented a mature version of the “*structuring grid*” used in the 20th century. The grid remained invisible under the final composition.

The postmodern grid is a visual element the designer manipulates. The grid remains in the final design. It is a visual element to express anti-rationality and randomness (Williamson 1986).

The Grid's resistance to time reveals its importance in representing composition factors to order visual elements. The purpose of this layout tool remained unchanged: activating the composition space to serve the design concept. Breaking the grid occasionally followed changes in perceptions and practices in visual organization.

Designers create representations to guide the manipulation of visual properties such as color, typography and layout. These representations summarize information about the visual property, and help experiment with it to express their visual concept.

2.6 THE VISUAL COMPOSITION PROCESS

In this section, I identify visual composition as a reflective practice (Schon 1984), where designers know more than they say. This practice also depends on what happens at the time of creating the visual composition. Thus, understanding this practice requires observation of how practitioners create their compositions in situ.

2.6.1 *Visual composition is a reflective practice*

Schon identifies a set of practices where skillful professionals know more than they say. These practices include design, city planning, engineering, management, law, education, psychotherapy, and medicine (Visser 2010).

Reflective practice is an activity where: *“doing and thinking are complementary. Doing extends thinking in the tests, moves, and probes of experimental action, and reflection feeds on doing and its results. Each feeds the other, and each sets boundaries for the other”* (Schon 1984).

Schon observes designers at work to describe the notion of *“reflection-in-action”*. Designers are in a constant conversation with design situations. They frame and reframe problems. Their attempts to solve their reframed problems lead to new discoveries, and thus to new reflection-in-action. This is a closed process of appreciation, action and re-appreciation (Visser 2010).

Schon also emphasizes the role of *“problem setting”*, where practitioners interactively name the things they will attend to and frame the context in which they will attend to them (Visser 2010). Reflective action involves creating a design world. Practitioners are in a constant conversation with the materials of the design situation.

Visual properties are part of the properties of the design materials reflective practitioners interact with. Each practitioner creates a personal design world, and a unique appreciation of the properties of the design materials (Schon 1992). Schon gives an example from one of his studies with architects. Petra, an architecture student, describes her design process around the blueprint she created for a school building (Fig. 25): *“I had six of these classroom units but they were too small in scale to do much with. So I changed them to this more significant layout (The L-shapes)”* (Schon 1992). Petra based her design move on a subjective judgment: *“too small in scale”*. This judgment is related to two visual properties of the prototype: scale and size. Petra changed

these two properties and created a new prototype. She again judges the organization of the new prototype based on its visual properties: “a more significant layout”, and decides it is convenient.

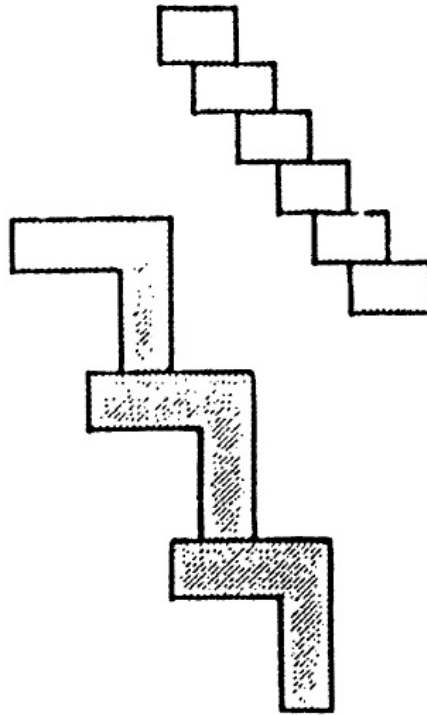


Figure 25: Petra changes the initial blueprint based on two visual properties: scale and size. (Schon 1992)

In reflective activities, the practitioner establishes a constantly evolving dialog between a prototype and a site (Schon 1992). The practitioner selects an initial prototype from her repertoire according to the constraints and possibilities of the site and the design task. Then, through reflective action, the prototype and site both transform based on the practitioner’s judgments in each step. These judgments are subjective, and they are often related to visual properties of the site or of the prototype.

Schon describes design work in a view similar to Suchman’s situated action (Suchman 2007). He considers design knowledge to be: a knowledge-in-action revealed only when designers are actually designing (Schon 1992).

2.6.2 *Manipulating visual properties in situ*

Suchman introduced “*situated action*” to emphasize the impact material and social circumstances have on any course of action (Suchman

2007). Purposeful action includes: plans, situated action and recollected reconstructions. Suchman uses the example of riding rapids in a canoe. One might come up with a detailed plan for the descent. However, when responding to currents and handling the canoe, during the actual descent, you abandon the plan and rely on any embodied skills available to you (Suchman 2007). Your recollection of the ride afterwards might be different from what actually happened. Plans, situated action and recollections involve different representations of the action.

Rules and procedures might be involved in the action's representation when creating plans. The action, when it occurs, is transparent: it is a response to the current circumstances. Rules might appear when the situated action becomes problematic. The action, which is not procedural by nature, accounts then for rules and procedure. I believe a similar phenomenon takes place when creating compositions. Designers know composition rules and guidelines. Procedures help them select the strategies they want to use. However, decisions of how the final composition will look like take place at the time of creation. For this reason, probing for the details of how designers create compositions provides more exhaustive accounts of their practices.

Designers know the rules and procedures that define and guide creating visual compositions. They also respond to the circumstances of the situation as they create these compositions. It is thus important to know the principles designers use to establish their composition plans. It is also important to identify practices of creating visual compositions in situ.

Designers know more about the task than they say. Their actions involve moving from one prototype to another in a closed cycle. They first use the visual properties of the prototype to make a qualitative judgment of the prototype and the design situation. Then, based on this judgment, they decide to act on some visual properties of the prototype to transform it, along with the design situation.

2.7 SUMMARY

Based on our empirical findings, and on the design principles of reification, polymorphism and reuse, I propose a framework for the manipulation of visual properties. I argue that visual composition tools should support the creation of personal, rather than generic representations of these properties. Also, designers should be able to change these properties directly in the composition. They should be able to combine and link them. Finally, they should be able to define how these properties evolve in time, across several compositions, or in space, across components of the same composition.

Perception, expertise, and design guidelines affect how designers create compositions. First, perceptual organization identifies different types of visual properties. Emergent properties depend on the current organization of the perceived object, while inherent properties relate to its components. Distinctive features are the minimal set of visual properties an observer needs to differentiate an object. Perceived affordances are the manipulation possibilities observers see depending on their action capabilities. Then, composition rules and guidelines encourage designers to use specific visual properties to ensure unity and variety, or to create emphasis and balance in their compositions. I focus on dedicated guidelines that help designers use three visual properties: color, typography and layout. Finally, I describe visual composition as a reflective practice. Designers know more than they say about their composition choices. Creating a visual composition also responds to factors from the current situation, and depends on the designer's visual concept. For this reason, observing designers as they create and organize visual content provides more complete insights about their composition practices.

RELATED WORK

This chapter discusses how composition tools' support the reification of visual properties. The desktop metaphor affected how visual properties appear in desktop composition tools. The Xerox Star and the Apple Lisa both represented visual properties based on the desktop metaphor. They use property sheets: dialog boxes to access properties when the user asks for them.

Early direct manipulation systems, such as Sutherland's Sketchpad and Kay's Dynabook proposed unique visions for changing object properties. Direct manipulation principles affected how applications support composition tasks, but focused on graphical objects. These principles did not propagate to object properties. More recent composition tools, designed for experts and novices, suggest different approaches to manipulate visual properties for composition tasks. I discuss how these approaches support the reification of visual properties into first-class interactive objects.

3.1 DIRECT MANIPULATION

Early direct manipulation systems such as *the Sketchpad* and *the Dynabook* suggested directions for representing properties in graphical user interfaces. The *Xerox Star*, and later commercial systems, applied *progressive disclosure* to properties. These properties are details hidden in dialog boxes accessed on demand (Johnson et al. 1989).

3.1.1 Sketchpad

Sutherland (Sutherland 1964) demonstrated early concepts of direct manipulation in the *Sketchpad* system (Fig. 26). Using a light pen, *the Sketchpad* supports direct manipulation of graphical objects. This manipulation includes grabbing, resizing, rotating and moving graphical objects.

The *Sketchpad* supports changing three visual properties: orientation, size, and position. Manipulating one of these properties occurs through constraining specific variables. For example, a user can define constraints to make a line horizontal (orientation), and two inches long (length). Figure 27 shows the chosen display for constraints in the *Sketchpad*. In this example, three sets of digits display the same scalar value: -5978. Users can change the size, orientation or position of the digits without changing the scalar value. The *Sketchpad's* design requires pointing the light pen at the abstract display: # to change the

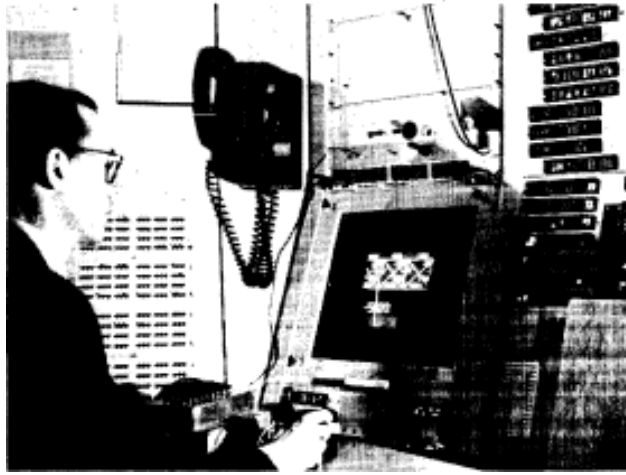


Figure 26: Ivan Sutherland using *Sketchpad* (Sutherland 1964)

scalar value. This would then change this value for all three sets of digits. The figure also shows two constraints that could apply independently of each other. One constraint applies to the scalar value and the other to the position of the digits.

The *Sketchpad's* design relies on a fundamental principle: constraints should be visible when needed. Constraint visibility is important because it reveals the existence of a constraint, and helps the user aim at it with the light pen (Sutherland 1964). Users can display constraints by turning on a toggle switch. When they turn off the toggle switch, only graphical objects -lines, circles, text, digits, and instances- appear on the screen.

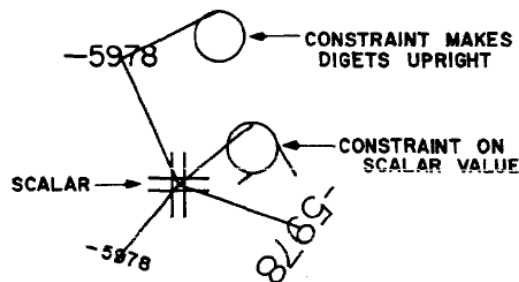


Figure 27: Three digits representing one scalar value in *Sketchpad*

Sketchpad's graphical manipulations were soon reused for 3D drawing in the MIT Lincoln laboratory (Johnson 1963). *Sketchpad* is also an early demonstration of using a pointing device to manipulate visible graphical objects on a screen.

Sketchpad supports manipulating size, orientation and position through constraints. These constraints are not always visible. They are displayed on demand, when the use wants to change them.

3.1.2 *The Dynabook*

In 1972, Kay (Kay 1972) introduced the *Dynabook*, a vision of the computer as an active medium for thought and creation. Kay describes the personal computer:

“ It would be both a medium for containing and expressing arbitrary symbolic notions, and also a collection of useful tools for manipulating these structures, with ways to add new tools to the repertoire.” (Kay 1972)

The *Dynabook*, like the *Sketchpad*, demonstrates notions of direct manipulation. It also advocates the importance of putting these notions in the hands of ordinary users. Kay gives the example of a drawing painting system (Fig. 28), programmed by a child. In this system, pictures are constructed from geometric shapes, using a menu of commands for creating regular polygons. These pictures are created by directly manipulating the polygons’ following visual properties: position, size, color and line width (Kay 1972).

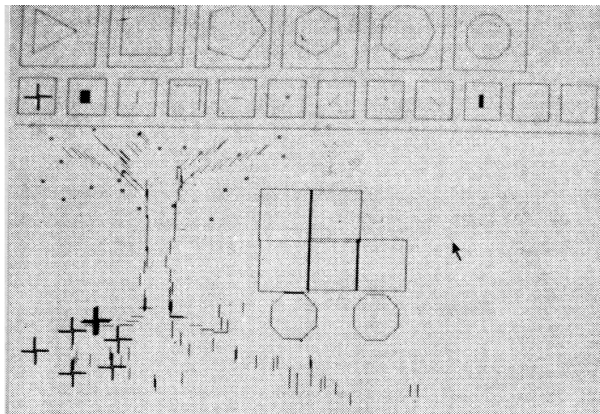


Figure 28: One of the first painting tools designed and implemented in Smalltalk by a twelve-year-old girl (Kay and Goldberg 1977)

The *Dynabook* also offers the possibility of creating tools that have interactive visual properties. For example, *Dynabook* pens can draw with ink of different colors, and with different thickness values. Also, a brush allows creating halftones depending on the size, the shape, and the velocity of the brush (Kay and Goldberg 1977).

Kay argues personal computers can be useful for a large audience if they manage to combine power and flexibility. Power comes from anticipating different use cases. The personal computer, like the car or the television, will present many functionalities. But unlike these

items, it will also offer the opportunity of taking unanticipated paths. Like paper or clay, the personal computer should offer many dimensions of possibilities. Klay presents two principles to design powerful and flexible personal computers. First, users should be able to describe their specifications. They should have a general medium to communicate their desires. Second, they should be able to create their own tools based on these specifications. They should have a set of starting tools, and the possibility to create new tools as needed (Kay and Goldberg 1977).

In this vision, users define the properties relevant to their composition tasks, and how they want to change them. In the previous example (Fig. 28), the child's designed interface included the properties of size, position, color and line width. The interface also included tool properties, such as the brush size, thickness, and velocity.

3.1.3 The Xerox Star

The *Xerox Star* design emphasizes recognition over recall, and seeing over remembering. One of the Star's design principles makes *the system's state always reflected on the display*. Another principle relies on *Object Properties* to vary the appearance of objects of the same type. Object properties have different types themselves. For example, "the **Size** property might be a number or a choice from a menu, and the **Justified** property of a paragraph is either ON or OFF" (Johnson et al. 1989).

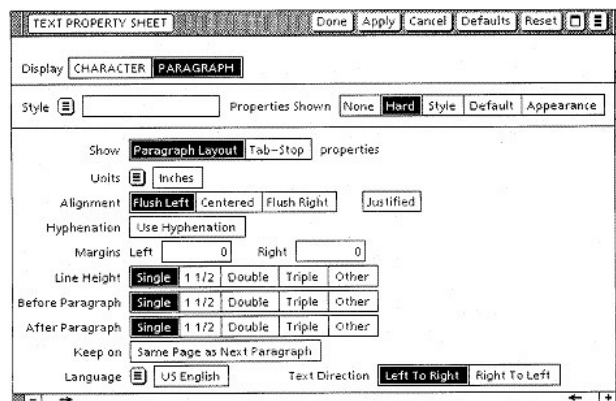


Figure 29: Progressive disclosure: The *Xerox Star*'s property sheets display object properties upon user request (Johnson et al. 1989)

The *Xerox Star* recognizes object properties as graphical entities in the interface. It also suggests design strategies to manipulate them. The *Xerox Star* uses the principle of *progressive disclosure* to represent visual properties. Progressive disclosure calls for hiding details from

users until asked for. The *Xerox Star* Property Sheets (Fig. 29) apply this principle to display object properties and set their values. Properties are details the *Star* hides from users, and displays them progressively, as the user selects the right fields in the Property sheet dialog box.

3.1.4 Direct Manipulation Principles

Shneiderman defined three concepts of direct manipulation (Shneiderman 1983):

- Continuous Representation of the Object of Interest
- Physical actions instead of complex syntax
- Rapid, incremental, reversible operations whose impact on the object of interest is immediately visible

These principles were demonstrated in early systems such as *Sketchpad*, the *Dynabook* and the *Xerox Star*. Later graphical user interfaces adopted these principles for graphical objects as targets of manipulation. These systems did not explicitly apply direct manipulation principles to visual properties. The *Sketchpad* links changing properties to constraints, based on a principle Sutherland calls “Draw first, and fix up later” (Johnson 1963). The *Xerox Star* recognizes the importance of visual properties, but suggests a rather indirect design choice for manipulating them. The *Xerox Star* property sheets do not follow the first principle: they represent visual properties only when the user asks for them. Setting and modifying property values happens in dialog boxes. The result of the manipulation is then visible only after closing the property sheet. The *Dynabook* is the closest to supporting direct manipulation principles for visual properties. For example, in the systems children created, they could directly change color and trace thickness on the composition (Kay and Goldberg 1977). These are examples where users created their own tools and defined their own specifications.

Direct manipulation principles affected the manipulation of objects in composition tools. Properties are details hidden and displayed on demand. The desktop metaphor also affected how properties appeared in composition tools. Properties did not have independent representations. They are inferred from users’ actions on the objects.

3.2 THE DESKTOP METAPHOR

Several commercial graphical user interfaces such as the *Xerox Star*, the *Apple Lisa*, and the *Apple Macintosh* relied on the desktop metaphor. The Macintosh human interface guidelines stated:

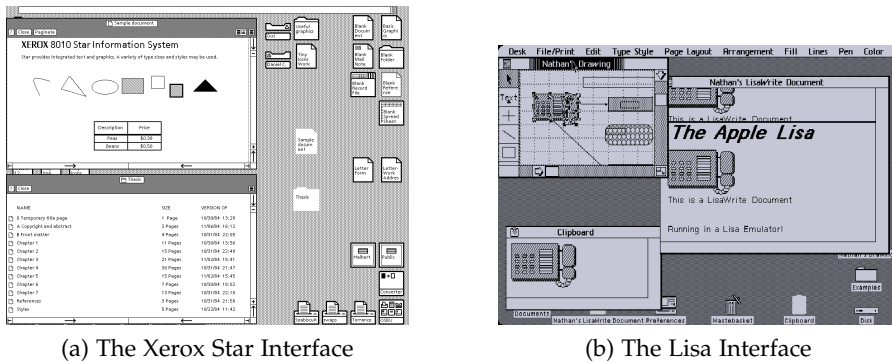
“You can take advantage of people’s knowledge of the world around them by using metaphors to convey concepts and features of your application. Use metaphors involving concrete, familiar ideas and make the metaphors plain, so that users have a set of expectations to apply to computer environments.” (Apple Computer, Inc 1992).

Halasz and Moran addressed the concern of using analogies to teach and design computer systems (Halasz and Moran 1982). Analogical models create a complete mapping between a known and an unknown complex system. This mapping also includes the conceptual structure of the systems: objects, relations, and operations. The two systems carry similar tasks, but familiar concepts from the structure of the known system might be fundamentally inappropriate for the unknown system. Halasz and Moran propose replacing analogical reasoning with a conceptual model of literary metaphors. According to Lakoff and Johnson, literary metaphors are natural, ubiquitous and grounded in everyday experience (Lakoff and Johnson 1980). Blackwell uses the following working definition of a metaphor: *“a representation created to help the user understand the abstract operations and capabilities of the computer”* (Blackwell 2006).

The desktop metaphor is a design approach based on the analogy between manipulating information in a computer system and manipulating physical objects on a desktop. The *Xerox Star* uses the desktop metaphor to create a visual representation of physical objects in the office: *“The Desktop is the principal Star technique for realizing the physical office metaphor. The icons on it are visible, concrete embodiments of the corresponding physical objects”* (Johnson et al. 1989).

The desktop metaphor presented an important consequence for computer systems: the display became a reality: *“Objects and actions can be understood purely in terms of their effects upon the display”* (Smith, Ludolph, and Irby 1985). For example, documents have two main visual forms on the display: icons and windows. These two visual representations imply different actions. Both icons and windows have a position. They are both movable on the display. On the other hand, windows have a size property that leads to an action of resizing while icons do not. This difference might seem trivial in today’s desktop interfaces: we expect to be able to resize a window, but not an icon. In both cases, the two visual properties, size and position, are inferred from the actions, but do not have any visual representation on the screen. The desktop metaphor focused on making graphical objects and actions visible on the screen. Visual properties on the other hand, are inferred from actions on the object’s visual form.

The *Xerox Star* and the *Apple Lisa* (Ludolph and Perkins 1998) automatically define which application opens a document. Starting from



(a) The Xerox Star Interface

(b) The Lisa Interface

Figure 30: Two Interfaces based on the desktop metaphor, centered on the concept of a document

the Macintosh (Williams 1984), the concept of applications adds an extra layer to the way the desktop metaphor was first articulated. Users now need to specify data files for each application. This is an example of use of the tools metaphor (Johnson et al. 1989), where applications are tools to access and edit data files. The desktop metaphor focuses on creating visual representations of physical objects in the office. Visual properties do not have a standard, defined representation on the screen. These properties are usually inferred from the user's actions on the object.

The desktop metaphor was introduced more than 40 years ago. It provided novel ways for tasks for which it was originally designed, such as routine office tasks. The adaptability of the desktop metaphor is questionable as interface users perform more diverse types of tasks (Tristram 2001). For example, composition tasks rely on changing objects properties in the document. I argue that this type of tasks requires rethinking the current representation of visual properties, still based on the desktop metaphor.

3.3 VISUAL COMPOSITION TOOLS

Some visual composition tools support specific expert practices as they create visual content. Other tools focus on facilitating visual composition for novices to create good, and accessible results. For these tools, I focus on the visual properties the tool represents, the manipulations it suggests, and the implications for experts or novices using the tool.

3.3.1 *Visual composition tools for novices*

Tools designed to support novices in visual composition tasks provide simplified views of visual properties, or perform additional configurations on behalf of the user. These tools include more generic

interface tools that can be used for visual composition.

3.3.1.1 *Automating Visual composition*

Systems that automatize visual composition often hide visual properties. They do not support full manipulation of these properties. For example, Reis suggests a multi-agent system based on visual composition styles created using rules and constraints extracted from expert practices. The novice user chooses and applies one of the proposed visual composition alternatives (Reis 2008).

AUTOMATIC COLOR SELECTION:

Many color tools are based on computational algorithms that hide the complexity of color manipulation. If users do not understand the underlying models, they have trouble using them to achieve specific effects. ACE (A Color Expert) (Meier 1988) addresses this by automatically selecting colors based on a model of the functional relationships among components of a design drawing. Wang et al.'s (Wang et al. 2008) colorization process lets users choose hues from a color wheel. Then, an expert system calculates optimal brightness and saturation values. These expert systems support clearly defined tasks but are less helpful for creative tasks in which users define their own rules and constraints. While they provide good-enough initial results, it remains difficult to modify the results.

AUTOMATIC LAYOUTS:

Many layout tools assist users in visual organization by creating automatic layouts. For example, DesignScape (O'Donovan, Agarwala, and Hertzmann 2015) makes suggestions during both the brainstorming and refinement phases of layout creation (Fig. 31). R-ADoMC (Jahanian et al. 2013) makes recommendations for magazine covers based on visual properties such as color themes and visual balance. Edge et al. (Edge et al. 2015), propose automatic alignment systems and systematic restyling of related objects to maintain consistency across slides. Piccoli et al. (Piccoli et al. 2011) propose an interactive system using physics principles to guide content organization on the page. These expert systems hide visual properties from the user. Instead, they perform automatic manipulation of these properties to propose visual organizations.

Other tools hide only specific visual properties, allowing users to change the remaining visual properties to create their layout. Automated responsive design (Colby 1992) creates semi-automated designs that adapt to devices' screen sizes and content selections (Schrier et al. 2008) (Kuhna, Kivelä, and Oittinen 2012). Sukale et al. (Sukale,

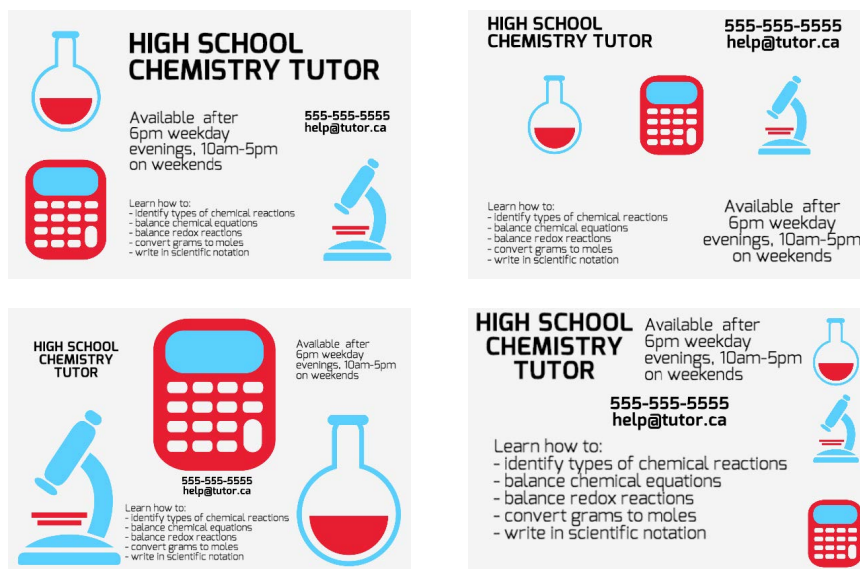


Figure 31: DesignScape: The system generates layouts in a variety of styles by sampling a low-dimensional parameter space learned from examples. (O'Donovan, Agarwala, and Hertzmann 2015)

Koval, and Volda 2014) also automate the adaptation of layouts based on users' proximity to the screen. In these examples, the proposed layouts automatically define the size of content elements. Adaptive grid-based document layouts (Jacobs et al. 2003) automatically choose and fill existing templates. Thus, the system automatically assigns positions to content elements. These systems reduce the number of visual properties users can change to suggest rapid layouts.

3.3.1.2 User interface tools for Visual composition

Several tools and systems rethink the traditional hierarchy of graphical elements, which focuses on graphical objects, and hides other elements, such as object properties and interface tools.

HABILISDRAW:

HabilisDraw (St Amant and Horton 2002) is a drawing application designed to explore the use of tools for drawing tasks. HabilisDraw includes a *Show Properties window*. Based on progressive disclosure (Johnson et al. 1989), this window provides access to the visual properties of drawn objects.

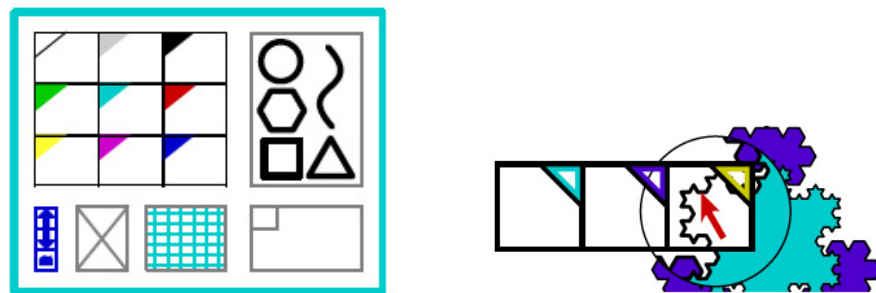
HabilisDraw considers tools as first class artifacts. The application provides access to tool properties. For example, the pen tool has a width property, and a color property. The ruler tool has two visual properties: length and angle. Tools' visual properties in HabilisDraw

can also be accessed and modified through *Property Sheets*.

HabilisDraw provides access to the visual properties of application tools. However, it does not suggest new ways of representing and manipulating these properties.

TOOLGLASSES AND MAGIC LENSES:

Toolglass widgets are interface tools that operate within the see-through interface. They appear on an additional layer between the application and the traditional cursor. Toolglass widgets are combined with viewing filters called Magic Lens filters (Bier et al. 1993). A single sheet includes multiple widgets (Fig. 32). These widgets provide customized viewing modes and multiple commands. The user can switch between these commands and viewing modes by repositioning the sheet.



(a) A sheet of widgets. Clockwise from upper left: color palette, shape palette, clipboard, grid, delete button, and buttons that navigate to additional widgets.

(b) An outline color palette over a magnifying lens.

Figure 32: Toolglasses and Magic Lenses (Bier et al. 1993)

The see-through interface, with toolglass widgets, represents visual properties in an additional layer, on top of the object. It also provides interactive tools to manipulate these visual properties through the sheet of widgets (Fig. 32).

SURROGATE OBJECTS:

Javed et al. (Javed et al. 2011) propose creating an additional representation: a surrogate object, to control the visual properties of a domain object. Any changes to the surrogate propagate to the domain object. In a drawing application based on surrogates, four types of surrogates illustrate how using a different representation to manipulate visual properties can support drawing tasks. The first type of surrogates is a shape, which exposes the following visual properties: line width, color, and transparency. The second type is a circle surrogate,

which exposes radius. The third type is a rectangle surrogate, which exposes height and width. The last type of surrogates is freehand, and provides access to individual vertices (Fig. 33).

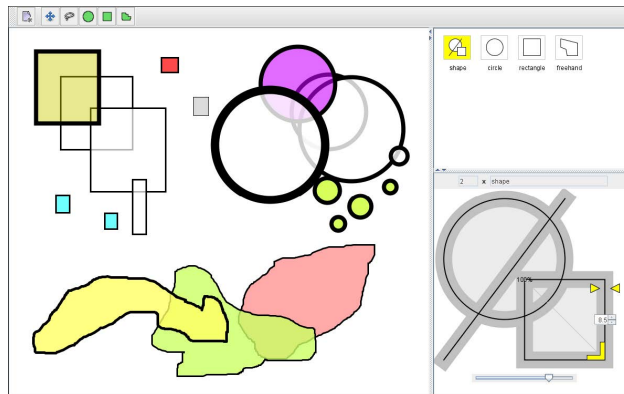


Figure 33: Vector drawing editor using surrogate interaction (Javed et al. 2011)

In this model, visual properties still depend on the visual object. Only specific properties are visible in the surrogate, and thus available for manipulation, depending on the object type. Designers define the visual properties they want to manipulate. These properties might not be part of the surrogate designed for this object.

OBJECT-ORIENTED DRAWING:

Object-Oriented Drawing replaces Wimp (windows, icons, menus, pointers) user interface elements with Attribute Objects: representations of digital content attributes as interface objects. This approach is based on considering attributes as independent objects, instead of parameters that determine the appearance of an object (Xia et al. 2016).

Object-Oriented Drawing proposes a novel representation of visual properties for touch interfaces. Visual properties are represented using a paper-like card. These cards can be attached to objects, as they might sit independently on the screen (Fig. 34).

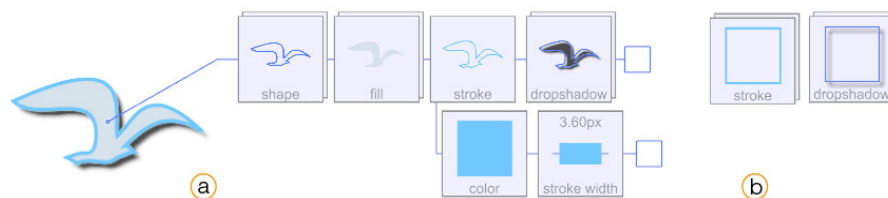


Figure 34: a. A collection of attribute cards, connected to an object.
b. Free floating attribute cards (Xia et al. 2016)

Object-Oriented Drawing extends physical direct manipulation of the attribute cards through: dragging, stretching and holding, using manipulation gestures. Specific gestures support manipulating visual properties to create new cards, delete cards, or detach them from objects (Xia et al. 2016).

These tools challenge existing representations of properties to support new devices, such as touch interfaces. They also show the importance of manipulating the visual properties of other graphical elements, such as interface tools. Other tools support manipulating visual properties for specific visual composition tasks. I focus on tools designed to support artists' and designers' visual composition tasks.

3.3.2 *Visual composition tools for experts*

Tools designed for visual composition experts base their design choices on experts' needs in specific phases of visual composition.

PARALLEL PIES:

Parallel Pies (Terry et al. 2004) is a user interface mechanism that supports design exploration through generating multiple alternatives. Users add variations of the current image. They use the Add variation command to duplicate the document, apply the current change, and insert the result in the workspace. Parallel Pies extends Side Views (Terry and Mynatt 2002b) to provide on-demand previews of multiple commands without changing the original data (Fig. 35). Side Views support experimentation with visual properties by displaying previews of potential future states (Terry and Mynatt 2002a).

Parallel Pies shows a variation of the whole original image to represent the change in visual properties. Like in the desktop metaphor, visual properties are still inferred from their effect on the object. The user compares different variations of the same image simultaneously, to see and change its visual properties.

ARTISTIC RESIZING:

Artistic Resizing supports the creation of multiple copies of the same object, at several key sizes (Dragicevic et al. 2005). Artistic resizing handles visual properties that have spatial semantics. For example, opacity is handled relatively to size: the smaller the object, the more transparent it appears.

Like Parallel Pies (Terry et al. 2004), artistic resizing represents visual properties by creating new instances of the graphical objects.

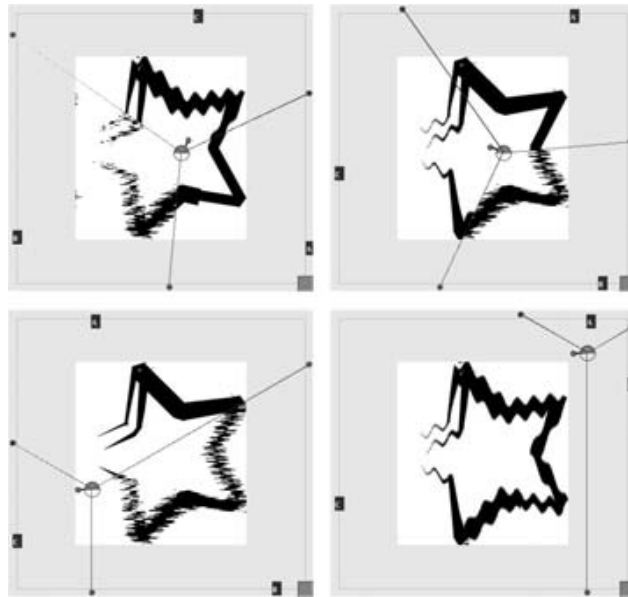
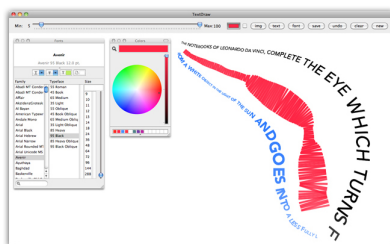


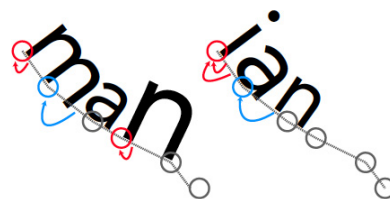
Figure 35: The Parallel Pies tool can be rotated and moved to selectively show all or part of a variation. A paddle extending from the hub controls the angle of rotation. (Terry et al. 2004)

TEXTDRAW AND TYPELS:

TextDraw is a prototype for gestural typesetting (Kirton 2011). It provides direct control over individual glyphs through gestures. *TextDraw* supports simultaneous manipulation of multiple visual properties. For example, typographers can change the size, color and position of a letter at the same time. Artists used *TextDraw* to create visual compositions based on gestural typesetting (Fig. 36.a).



(a) The interface for *TextDraw*. On the left are panels for choosing font and colour. On the right two lines of text have been drawn in blue and black, with a third currently being drawn in red



(b) An example of 2 words placed on the same baseline. Notice the letters a and n are different sizes between the word man and the name ian. This is because the letters m and i have different widths.

Figure 36: *TextDraw* and *TypeIs* use gestures to draw with type (Kirton 2011)

TypeIs extends *TextDraw* to provide better storage and rendering capabilities. *TypeIs* also generates weighted baselines: visual represen-

tations of a property of type: visual weight (Fig. 36.b).

TextDraw and *Typels* support new ways for visualizing and manipulating the visual properties of typefaces. Their approach to visualizing properties is based on generated baselines for individual characters. The two tools support the manipulation of type properties through capturing differences in users' input gestures. While they use baselines for type properties, the two tools still rely on a traditional color picker for color selection, and a dialog box for font selection (Fig. 36).

EXPLICIT AND IMPLICIT STRUCTURES:

Based on contextual interviews with designers, Hoarau and Conversy (Hoarau and Conversy 2012) designed a drawing application where visual properties are visible next to the main workspace. Users can change visual properties by dragging a sample from the property sheet to the object on the workspace (Fig. 37). Users can also create

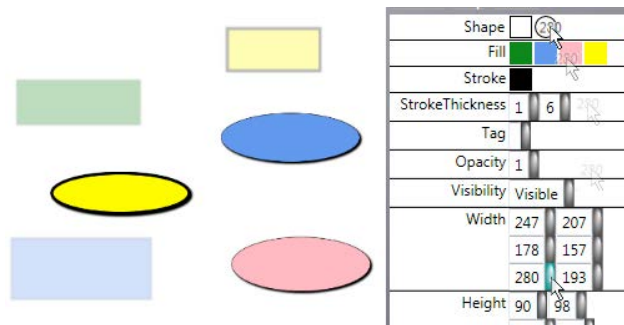


Figure 37: The user drags the “width: 280” shared value and drops it on the “shape: circle” shared value. All circles in the selection now have a width set to 280. (Hoarau and Conversy 2012)

links between visual properties of different graphical objects. Users create links by clicking on a property, drawing a link, and dropping it onto another object (Fig. 38). The concept of “cloning” supports creating a new object from a previously existing object. Unlike a copy, a clone references the visual properties of the original objects. Thus, changing the visual properties of the prototype (the original object), changes the properties of the clone.

The application suggests an augmented property sheet, also based on the concept of progressive disclosure (Johnson et al. 1989). These property sheets display all possible values of categorical visual properties. Properties with numerical values appear as blank text fields. Augmented property sheets enhance the manipulation of visual properties by supporting the selection of objects with the same visual property (Fig. 37). The user can then apply a second visual property to

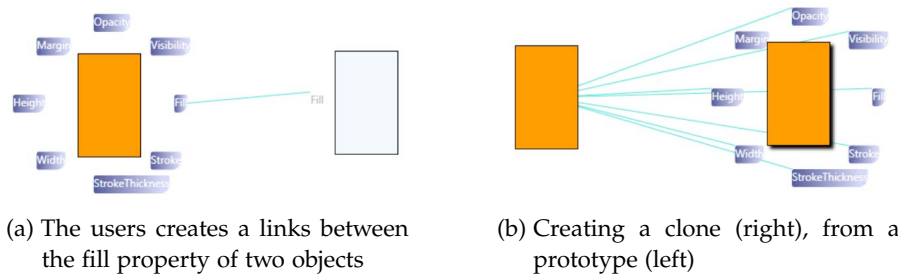


Figure 38: Creating links between visual properties (Hoarau and Conversy 2012)

these objects from the property sheet.

KITTY:

Kitty is a sketch-based tool for creating interactive illustrations (Kazi et al. 2014b). Kitty supports creating functional relationships between objects through direct manipulation. The system focuses of spatial and temporal properties of the scene's visual entities.

Kitty extends the Draco (Kazi et al. 2014a) system, which uses patch and motion properties to encode collections of objects and their motions. In addition to this animation component taken from Draco, Kitty adds an interactivity component. Interactivity in Kitty is based on a framework where objects and collections have attributes. These attributes define their visual properties such as shape or color.

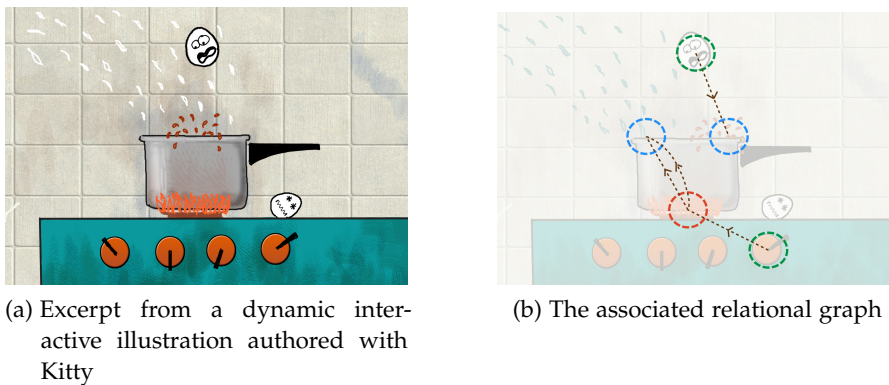


Figure 39: Artists can create links between elements' visual properties using a relational graph (Kazi et al. 2014b)

Artists can use Kitty to create animated objects and textures. They can create links between the visual properties of entities within an illustration. Creating links requires creating a relational graph in the graph mode (Fig. 39). Kitty's framework uses visual properties for

sketching visual entities within the illustration. A graph model captures links between the spatial and temporal properties of visual entities.

TOOLS FOR DIRECTLY MANIPULATING VISUAL PROPERTIES

Direct manipulation mechanisms, such as copy-paste, drag and drop, and undo, are ubiquitous in graphical user interfaces (Myers 1998). New features extend these mechanisms to support new composition tasks.

Based on the observation of users' selection practices, Paint Selection progressively solves a series of local optimization problems to match users' directions, without sacrificing usability and selection quality (Liu, Sun, and Shum 2009). Aquamarine (Fig. 40) uses a script model for selective undo. It allows users of a painting program to select and undo specific operations in the past, while keeping the remaining operations unchanged (Myers et al. 2015). Paint Selection and Aquamarine both extend the "undo" command. They represent parts of the graphical object to keep traces of previous visual property manipulations.

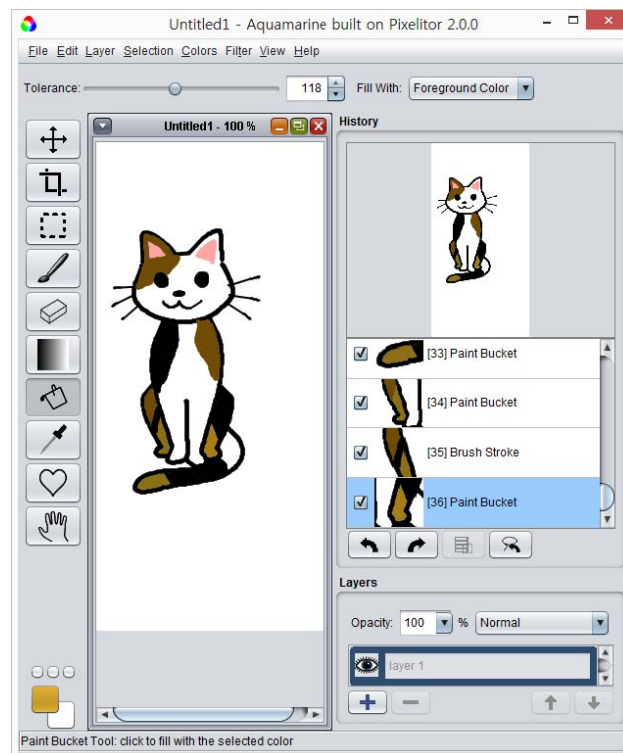


Figure 40: Aquamarine shows support for the history of visual property manipulations (Myers et al. 2015)

COLOR MANIPULATION TOOLS:

Several web-based tools allow users to experiment with different color combinations. Adobe Color ¹ lets users combine colors around a particular theme: they can manipulate multiple color swatches sampled from images, or generate palettes based on color rules, e.g., color triads or complementary colors. Other tools preselect sets of colors to address specific requirements. Color Brewer (Harrower and Brewer 2003) helps users choose effective color schemes for coloring maps, based on specific properties of the data being represented and human color perception. iWantHue ² helps users generate palettes with optimally distinct colors. Both have the advantage of simplicity, but with a very limited scope. Several recent tools explore innovative ways of manipulating color. For example, Histomages (Chevalier, Dragicevic, and Hurter 2012) allows users to edit images by modifying a histogram of the colored pixels within the image. Users can select and change subsets of colors, such as turning the sky from shades of blue to shades of orange (Fig. 41). Pouli and Reinhard (Pouli and Reinhard 2010) demonstrate how to transfer color by progressively matching a histogram to a target image's color, and Meier et al.'s Interactive Color Palette Tools (Meier, Spalter, and Karelitz 2004) offer additional techniques for interacting with color. Although these tools offer promising new techniques for manipulating color, they remain isolated from mainstream authoring applications.

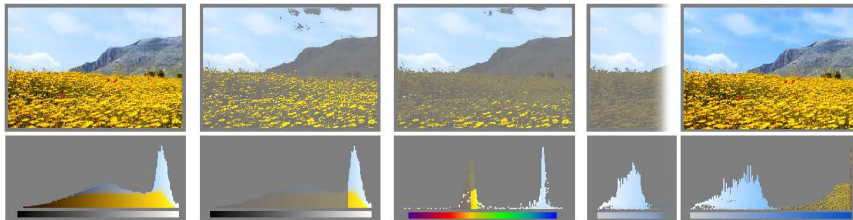


Figure 41: (a) the image is duplicated and its pixels rearranged into a lightness histogram; (b) bright pixels are selected with the rubber-band selection tool; (c) all pixels are rearranged into a hue histogram and yellow pixels are filtered out; (d) the sky is enhanced by resizing the selection on the saturation histogram. (Chevalier, Dragicevic, and Hurter 2012)

ALIGNMENT TOOLS:

Several tools support designers in creating visual organizations. Adobe Comp ³ allows designers to draw rapid initial layout ideas before

¹ <https://color.adobe.com/>

² <http://tools.medialab.sciences-po.fr/iwanthue/>

³ <https://www.adobe.com/products/comp.html>

moving to specialized applications. Systems such as Adaptive Ideas (Lee et al. 2010) help create website layouts by sampling elements from examples. DÉCOR (Sinha and Karim 2015) is a recommendation tool that helps designers adapt their layouts for a wide range of screen sizes. Adobe Edge Reflow⁴ also helps designers envision their layout for various devices using media query breakpoints.

Other tools support specific tasks in the process of creating a layout. For example, alignment tools suggest new ways to perform this tasks. For example, GACA (Xu et al. 2015) is a group-aware alignment technique that helps professionals deal with complex alignments. Moulder and Marriott (Moulder and Marriott 2012) propose a machine learning approach to solve line-breaking issues. NEAT (Frisch, Langner, and Dachsel 2011) and Grids and Guides (Frisch et al. 2011) suggest representations of the alignment visual property. NEAT introduces *Cross and Snap*, an interaction technique to create guides and bind objects to this guide at the same time. Users can change the visual properties of the guide. They can change its size, angle and position. StickyLines (Felice et al. 2016) represent alignment and distribution using *Guidelines*.

3.4 VISUAL COMPOSITION TOOLS SUPPORT OF THE REIFICATION OF VISUAL PROPERTIES

Several visual composition tools recognize the importance of manipulating visual properties for composition tasks. Some tools incorporate ideas related to the reification of visual properties. These ideas are rarely articulated as design principles within the tool. Also, these tools support only some aspects of the reification, depending on the constraints of specific tasks or media. I use the framework in section 2.2.2 to discuss how current visual composition tools support the reification of visual properties. I discuss the approaches these tools suggested to represent and manipulate visual properties.

INFERRED VISUAL PROPERTIES

Several tools represent visual properties through their effect on the object. They create and maintain alternatives the user can revisit. Side-views (Terry and Mynatt 2002b), parallel pies (Terry et al. 2004) and GEM-NI (Zaman et al. 2015) support exploring multiple design alternatives. They capture how visual properties evolve over time, through saving the state of the whole object. This approach supports saving the contextual information surrounding visual properties. They do not support creating personal representations of visual properties.

⁴ <https://www.adobe.com/products/edge-reflow.html>

They do not support creating links or rules that describe how properties evolve across alternatives of the same object.

ENHANCED PROPERTY SHEETS

Several tools propose enhanced versions of the traditional property sheets, introduced in the star (Johnson et al. 1989), and adopted in most commercial applications. For example, Habilisdraw suggests property sheets to manipulate the visual properties of interface tools. Augmented property sheets (Hoarau and Conversy 2012) show visual properties next to the composition. These properties are always visible. Users can select them. They can change all objects with a specific value for a visual property in one manipulation. Augmented property sheets support creating links between visual properties. They do not support creating personal representations of these properties, or creating rules to define how they evolve.

This approach suggests new types of property sheets, while preserving the representation and basic interactions of the traditional property sheet. This approach maintains consistency with existing visual composition tools. It also retains some of the challenges of the property sheet. Visual properties are not accessible in the composition. Users cannot create personal representations of these properties. Also, this approach keeps the dependency between properties and objects. Property sheets depend on the currently selected object or objects. Thus, links and rules between visual properties are difficult to create.

SEE-THROUGH INTERFACES

See-through interfaces suggest representing visual properties on top of the composition. They provide contextual access to the properties of interface elements. For example, toolglass and magic lenses (Bier et al. 1993) include visual properties, and tools to manipulate them, in a transparent layer that floats on top of the object. This approach supports manipulating visual property in the composition. It does not support creating personal representations of visual properties, establishing links between them, or rules to define how they evolve.

DEDICATED VISUAL PROPERTY OBJECTS

Some visual composition tools suggest creating new, dedicated representations of visual properties. These tools are the closest to reifying visual properties into first-class interactive elements. For example, Attribute cards (Xia et al. 2016) and surrogate objects (Javed et al. 2011) are independent representations of visual properties. Users can create, delete and manipulate the properties these cards represent. They can link these properties. Attribute cards and surrogate objects do

not support creating rules for how a visual property evolves, or create personal representations of visual properties.

Reification of visual properties				
	Personal Representations	Links between properties	Rules for evolving properties	Manipulation in the composition
Inferred properties	Side Views		Paint selection	Artistic resizing
	Parrallel Pies		Aquamarine	
Enhanced property sheets	HabilisDraw		Augmented Property Sheets	
See-through interfaces	Toolglasses and Magic enses			
Dedicated objects	Surrogate objects		Attribute cards	Kitty (relational graph)

Figure 42: Composition tools' support of the reification of visual properties.

3.5 SUMMARY

Visual properties are part of graphical user interfaces. Early visual composition systems based their design choices on direct manipulation principles, and on the desktop metaphor. In these systems, visual properties are considered details, and inferred from the user's action on the object. More recent visual composition tools suggest different approaches for manipulating visual properties. Tools designed for novices often hide all or part of an object properties. Novices can choose from a set of suggestions the system generates. Expert visual composition tools focus on specific tasks.

Existing approaches for manipulating visual properties include inferring the properties from their effect on the object, through suggesting several alternatives of the same object. They also include suggestions of enhanced property sheets, that support more ways of manipulating visual properties. The see-through interface approach suggests creating an additional layer, on top of the object, to manipulate its properties. Finally, a fourth approach creates an independent representation for visual properties in the composition. This latter is closer to the reification of visual properties.

MANIPULATING VISUAL PROPERTIES IN A DESIGN APPLICATION

Graphic design applications suggest different types of tools to support visual composition tasks. We want to know the tools and strategies graphic design professionals choose for composition tasks. This chapter presents a comparative structured observation study of the tools and strategies graphic design professionals choose as they recreate visual compositions in a design application. Initial observations revealed potential factors that might affect the tools and strategies used for visual properties manipulation. We found that graphic designers prefer manipulating properties directly in the document. They use tools that provide direct access to these properties. Designers also take advantage of visual properties to find innovative strategies for completing visual composition tasks. Participants perceived some strategies as Clever Tricks and others as Bad Hacks. In a follow-up study, we found that this perception holds for strategies used in other applications.

4.1 INTRODUCTION

Over the past 30 years, graphic design applications have introduced new features to better support visual composition. This resulted in phenomena such as software “bloat” and creeping featurism. Software “bloat” occurs when the cost of adding new features outweighs the benefit (McGrenere and Moore 2000) (Kaufman and Weed 1998). Creeping featurism happens when adding new features leads to a less homogeneous interface, as experienced by the user (Hsi and Potts 2000). Increasing the number of tools might lead to problems such as bloated, less homogeneous user interfaces.

Introducing new features did not introduce new ways of manipulating visual properties. The new feature types resemble those introduced in the Macintosh in 1984 (Williams 1984) (Myers 1998). They still include “a menu bar and overlapping windows, and a limited set of interaction techniques, including widgets (menus, buttons, dialog boxes, scroll-bars), drag&drop and copy-paste.” (Beaudouin-Lafon 2004)

The Adobe creative suite introduced advanced tools to better support visual composition tasks. For example, the official release of Adobe Illustrator CS6 describes the benefit of the new pattern feature saying:

“Pattern creation and editing tasks have been simplified, so you can save hours of repetitive and tedious work. The new Pattern

Options panel provides you with an easy set of options to experiment and modify your design until you get the pattern of your choice.”

The pattern panel mentioned above, and the gradient on stroke feature were both introduced in Adobe Illustrator CS6 ¹. Both are dialog-box widgets with a list of visual properties. Users need to modify the values of these properties to create the final result. This type of tools already existed in the Xerox Star property sheets (Johnson et al. 1989) (Fig. 43).

We are interested in the types of tools graphic design profession-

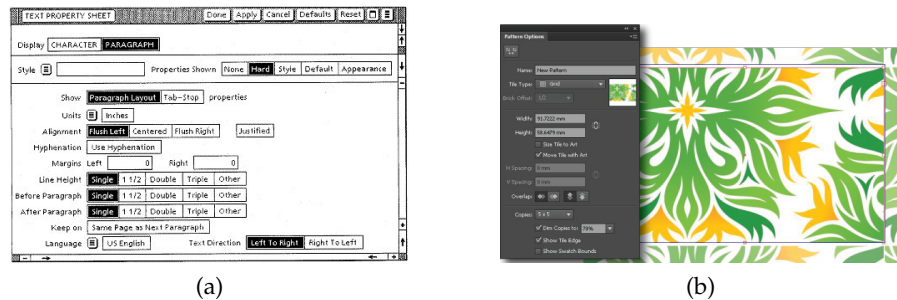


Figure 43: Property Sheets in the Xerox Star and in Adobe Illustrator Cs6

als choose as they create visual compositions. Tool choice provides insights about the tools professional designers find “*useful*” for manipulating visual properties. We are also interested in the visual composition strategies they follow.

We define a strategy as the tool or tools users choose for completing the visual composition task, and the detailed description of the steps leading to the final output. We want to know the visual composition strategies graphic design professionals choose, and the factors that affect their choices.

We run a study where we ask 12 professional graphic designers to recreate posters in a design application. We select a visual composition task involving heavy manipulation of visual properties. Participants manipulate visual properties such as color, size, repetition, alignment and layering. We are interested in the tools and strategies graphic design professionals choose when manipulating visual properties in a professional design application.

Our initial observations pointed at two factors that might influence the choice of visual composition strategies. The first is result quality, when users create high versus low quality results. The second is repetition, when they create results with one versus many occurrences.

We want to vary these factors to compare graphic design professionals’ strategy choices for visual composition tasks. We designed a

¹ <https://helpx.adobe.com/illustrator/using/whats-new-cs6.html>

comparative structured observation study. We asked 12 professional designers to recreate two isomorphic posters. We then compared tool and strategy choices across participants. We also identified the types of tools and strategies they use more often. This study design allowed us to compare tool use and strategy choices across participants while maintaining contextual details of individual user strategies.

4.2 CONTEXT

4.2.1 *Designers's composition practices*

Experts and novices perceive and perform tasks differently. They differ in the granularity with which they view the constituents of a particular problem or task. Novices are more attentive to low-level details. Buxton gives the example of finding a particular character on the keyboard or remembering the name of a command. These activities require problem solving, which diverts the novice's cognitive resources from the central problem at hand (Buxton 1986).

Graphic design professionals want to find unique approaches for each design problem. Berryman, a designer and design teacher explains how each design problem presents unique challenges. Yet, some commonalities do help designers structure their approach to solve the problem (Berryman 1990). Terry and Mynatt describe design problems as open-ended tasks. They find that users experiment to better understand the problem and their available options. They generate alternative strategies to approach the problem from multiple angles. They also reflect on their progress to inform their future actions (Terry and Mynatt 2002b). Designers perform these activities insitu. They choose a convenient approach when confronted to a problem, without always being able to answer theoretical questions about what they would do in a similar situation (Visser 2010).

While these studies of graphic design professionals describe their practices, they do not specify the types of tools they use or how they use these tools for visual composition tasks. We want to know how graphic designers choose tools and visual composition strategies for manipulating properties in a design application.

4.2.2 *Approaches to study software use*

Major approaches investigating software use include quantitative, qualitative, and mixed methods. Laboratory experiments focused on comparing different applications or different formats within the same application. Galetta et al. studied error finding performance from different Lotus 1-2-3 spreadsheet presentation formats. They proposed

five formats: on screen or on paper, with and without formulas, and with formulas presented in each cell directly under each calculated value. Users found only about 50% of the errors across all presentation formats (Galletta et al. 1996). In another experiment, nine experts in Lotus 1-2-3 each created three spreadsheets. Although users were confident their spreadsheets were error free, 44% contained formulae errors (Brown and Gould 1987). This suggested the need for tools to debug spreadsheet formulae. Similarly, Saariluoma and Sajaniemi studied the time needed to learn a spreadsheet formula. They showed that users learned them faster when the surface appearance was congruent with the computational structure beneath. This suggested that spreadsheet tools should show how the surface maps to the computational structure (Sajaniemi and Pekkanen 1988). Baxter and Oatley compared learnability of two spreadsheet applications, Microsoft Excel and Wingz, a spreadsheet application on Macintosh. They found that expertise in creating spreadsheets affected learnability while the spreadsheet application brand did not (Baxter and Oatley 1991). These types of studies provide design recommendations based on the impact of specific experimental factors on users' performance.

Another type of studies includes in-depth interviews with software users. Ethnographic interviews with eleven spreadsheet users showed that specific properties of spreadsheet software enable cooperation. For example, dividing the interface into two programming layers allows distributing computational tasks among users with different levels of programming knowledge. Also, the spreadsheet's visual structure helps users share domain knowledge. The authors described studying software use as one of the *"topics that by their very nature cannot be identified under the controlled conditions of the laboratory"* (Nardi and Miller 1991). In another study, Hendry and Green ran interviews with ten spreadsheet users about their experience with spreadsheets. They found that users focused on the weaknesses of the model while debugging or interpreting spreadsheets. On the other hand, experts' description of the spreadsheet model emphasized its strengths (Hendry and Green 1994). These types of studies provide a contextual understanding of user practices based on rich stories extracted from interviews and observations.

Other approaches investigate specific aspects of software use. Sajaniemi and Pekkanen analyzed 135 spreadsheets from different fields. They found that only 5% of cells contained formulae. They concluded that most spreadsheets are computationally very simple (Sajaniemi and Pekkanen 1988). Napier et al. collected command usage data from experienced users of Lotus 1-2-3. They found that 5% of commands available in the application accounted for 85% of the total commands issued. Also, commands for copying and moving data and

for changing the appearance of the sheet, accounted for 45% of commands used (Napier et al. 1992). McGrenere and Moore used a triangulation of methods to study how users experience complex software. They identified two types of features. A first subset includes features that are not used or wanted by any user. The second type includes two sets which vary from user to user and are thus subjectively defined (McGrenere and Moore 2000).

We want to systematically compare strategy choices across graphic design experts, while maintaining the contextual details of each user choice. For this reason, we choose a comparative structured observational study. This design involves specifying factors that might affect user choices, while still investigating emerging patterns in users' strategies as they perform the visual composition tasks.

4.3 INITIAL OBSERVATIONS

Graphic design professionals create their strategies as they solve visual problems (Visser 2010). Their recollection of the tools and visual composition strategies might be different from what they actually do when creating the visual content. For this reason, we observe graphic design professionals as they complete composition tasks.

We want to identify the tools and strategy choices different graphic designers choose for the same task. For this reason, we define key factors for a more systematic study. We observe how three users perform the same visual composition task, in three applications. Figure 44 provides an example of different strategies for creating a shape. In Microsoft PowerPoint (Fig. 44.a), Alice draws two squares and creates a white one to hide their intersection. In Adobe InDesign (Fig. 44.b), Bob deconstructs each side to three smaller squares. He then groups them to create the "L" shape and duplicates it on the other side. In Adobe Illustrator (Fig. 44.c), Carol creates two identical squares and a smaller square on top. She then subtracts the intersection between the two initial squares. She duplicates and resizes one of the initial squares and places it inside the white one.

Our preliminary observations revealed that users might choose different tools or visual composition strategies depending on the tools available in the application's graphical user interface. Another factor that might affect their choices is the type of visual composition tasks they try to perform. For example, if they need to create a visual output with several occurrences, they might opt for a tool or a strategy that reduces the amount of manual repetition. The choice of tools and strategies might also depend on the output quality the user is looking for. Thus, they might choose different tools or strategies when they want to create an accurate output, to send to the printer; while they

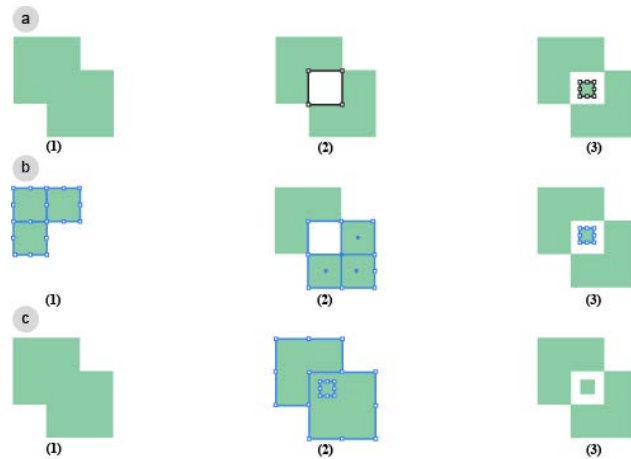


Figure 44: Three visual composition strategies for the same visual composition task.

might accept a more approximate output, when creating a first version of a document. Other possible factors include time and peer pressure, or the document destination medium. We focus on two factors: working on a visual output with one versus multiple occurrences, and on high versus low quality outputs. We use these factors to systematically compare users' choices of tools and strategies, as they create different output qualities and different visual composition tasks.

4.4 STUDY OF DESIGNERS TOOLS AND STRATEGIES FOR MANIPULATING VISUAL PROPERTIES

We are interested in the variety of strategies and tools graphic design professionals choose to complete visual composition tasks. We want to compare these choices and identify recurrent strategies or tools within a professional design application. We use comparative structured observation (section 1.3.3) to compare designers' strategy and tool choices for the same visual composition tasks.

We designed a [2x2] within-participant comparative structured observation study to extract and compare designers' strategies in visual composition tasks. Participants complete the design of two isomorphic posters. Each poster involves six visual composition tasks. They cover the values of two primary factors; the first factor is output quality: users complete one of the two posters with a high output quality instruction and the other poster with a low output quality instruction. The second factor is task type: composition tasks with a visual output occurring once, versus composition tasks with a repeated visual output. For each of these two factor values, we choose three composition tasks, supported by different tools within the application.

4.4.1 Participants

We recruited 12 professional designers (10 men, 2 women, aged 23-30), including four interaction designers, four designers and four product designers. All use Mac OS X and have extensive experience in performing visual composition tasks with Adobe Illustrator.

4.4.2 Hardware and Software

All participants used their professional laptop computers under OS X and their own installed version of Adobe Illustrator. We encouraged participants to use any input or output devices they usually work with. Ten participants used their computer trackpads. One participant used a graphic tablet and one participant used a mouse. Seven participants used the Creative Suite six (CS6), four the Creative Cloud and one the Creative Suite five (CS5). We used QuickTime player for screen and audio recording.



Figure 45: Initial and final versions of the two study posters

4.4.3 *Task*

Each poster includes six different tasks. The remaining visual elements already exist in the initial file (Fig. 45). Poster A (Fig. 45.a) involves six tasks: Replicating a text box in the background, highlighting text, aligning a text box in the art board center, creating a logo and copy-pasting text format. We provide participants with the color code they need to replicate and the indication that the format is similar. Poster B (Fig. 45.b) also involves six tasks: creating a square, creating a vertical pattern with squares, cutting squares anchor points, selecting multiple squares, copy pasting color and creating a logo.

4.4.4 *Procedure*

We ask each participant to complete the design of two posters in Adobe Illustrator: one poster with high output quality, for a client deliverable, and another poster with low output quality, for an initial version of the document. Within each poster, we expect the participant to perform two sets of tasks: three involve a unique and three involve a repeated visual output. Each participant performs 12 distinct visual composition tasks, within two isomorphic posters, counterbalanced for order and output quality across participants.

We provide starting files from an external drive. We do not impose any order for the six tasks. Participants have as much time as they need to complete the two poster designs. They can use only tools internal to their version of Adobe Illustrator, and only visual content within the current poster. They start with a warm up task that lasts two minutes, where we explain the study protocol. Then, they open the files from the external drive. We provide them with the task instruction, start the screen and audio recordings, and make sure the final files are properly saved once a poster design is complete. Following this initial training, we provide the participant with a printed version of the first poster in their condition and read the corresponding instruction: high or low output quality. Then, we ask the participant to describe out loud their steps and the tools or shortcuts they use. We proceed similarly for the second poster. We stop the screen recording when participants say they finished. We then ask them to fill out a final questionnaire. We debrief about the tasks before we stop the audio recording.

We log each participant's screen while performing the task, and keep an audio recording of the whole session. The initial questionnaire includes questions about the users' background in using Adobe Illustrator. The final questionnaire includes questions about their per-

ception of the overall difficulty of each poster, and asks for comments about the tasks they performed.

4.4.5 *Data Analysis*

We want to know users' tool and strategy choices for each of the study's 12 visual composition tasks. For this reason, we extract from each user's screen recording the strategy and tools used for each visual composition task. A strategy is a detailed description of how the participant proceeded to complete the task. Within each strategy, we identify the tools, describe the steps, and extract quotes and comments from the participant's audio recording. Two strategies are similar if they include the same steps using the same tools. For example, to cut 12 squares into triangles (Fig. 45.b), a user creates the visual output by cutting the anchor points using the pen tool for each of the twelve squares individually. In a different strategy for the same task, the user starts by creating a rectangle along the intersection line. She applies the background color to this rectangle, and uses the path finder to cut the rectangle and its intersection with the three squares. She then duplicates the result of cutting these three squares and rotates it for the remaining three occurrences. In this case, the two strategies include different steps and different tools. To create the umbrella shape, a user follows a strategy where she creates a large circle, followed by three smaller circles. She then places them inside the larger circle, and uses the path finder to extract the overlapping path. In a different strategy, she follows the same steps to create and place the circles. In the last step, she uses the pen tool instead of the path finder, to cut individual anchor points. In this case, although the two strategies share the same steps, they involve different tools for creating the same visual output.

4.4.6 *Results*

Participants used 41 different strategies to finish the 12 visual composition tasks in the two posters (Fig. 46). The visual composition task with the largest number of strategies is cutting the 12 squares into triangles in poster B.

4.4.6.1 *Manipulating visual properties in the composition*

Graphic design professionals manipulate visual properties directly on the graphical objects. We call strategies where users manipulate visual element directly in the document: direct strategies. Indirect strategies involve manipulating these elements through dialog boxes.

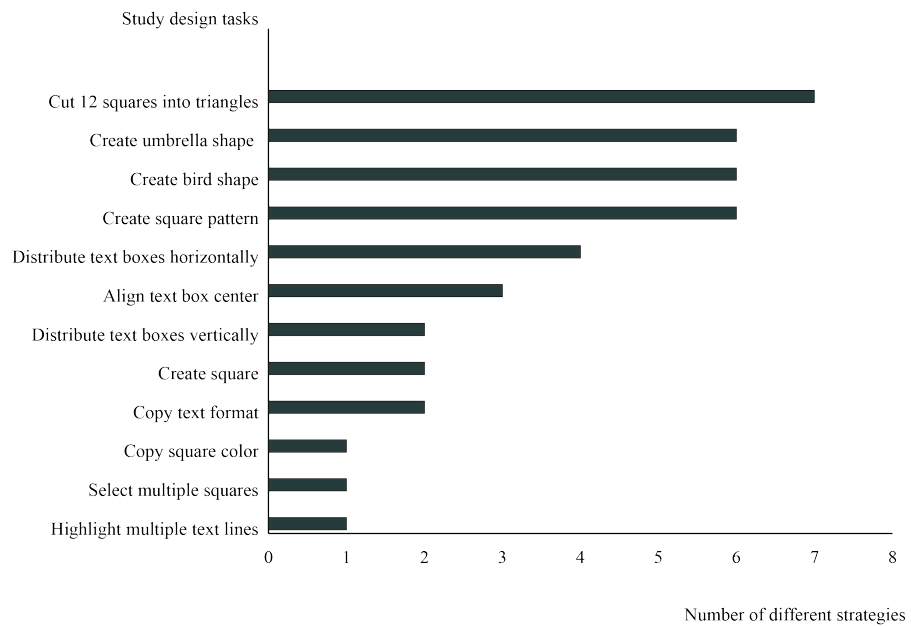


Figure 46: Number of different strategies per visual composition task

For example, to create a rectangle, a direct strategy is to select the rectangle tool, and draw the rectangle on the document. An indirect strategy involves selecting the rectangle tool, typing the desired width and height in the dialog box, and clicking OK. For all 136 extracted strategies, there is at least one direct strategy to complete the visual composition task. However, 88 out of 136 strategies have an equivalent indirect strategy. Across users' strategies, 96% (131 out of 136) are direct and only 6% (5 out of 88) are indirect strategies.

Repetition did not affect participants' choice of direct strategies. Almost all strategies participants used for repeated tasks (95%) were direct, even if choosing a direct strategy increased the number of repetitions. In one extreme, participants repeated the task for the smallest repeatable unit. We call these repeated action strategies. In this type of direct strategies, participants create the output for a single visual element and then replicate it to complete the number of occurrences. For example, eight participants duplicated a group of text boxes to create the vertical text box distribution in poster A. The second type of direct strategies participants used for repeated tasks is: modular strategies. These involve the creation of composite units participants reuse to complete the number of occurrences. For example, five out of seven strategies participants used to cut 12 squares into triangles in poster B include modular repetitions. In these strategies, participants cut a group of squares, and then reused this module instead of cutting squares individually. In the two remaining strategies for the same task, they used repeated actions where they cut each of the 12

squares.

Two participants finished a repeated task in one manipulation. We call these: single action strategies (Fig. 47). They involve setting parameters to create the repeated output in a single action. For example, to create the background text pattern in poster A, P9 and P10 modified the blend options and generated the 35 text boxes at once. Similarly, to create the square pattern in poster B, P10 modified the spacing, color and number of occurrences in the pattern option panel to generate the square pattern.

Participants used modular repetitions in more than half of the strate-

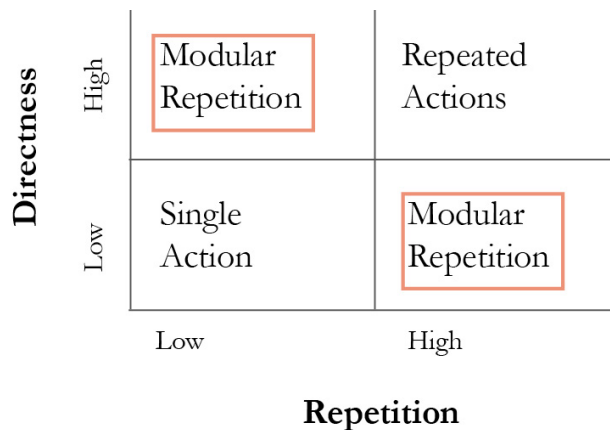


Figure 47: Modular repetition is a reasonable trade off between single action and repeated actions.

gies for repeated tasks (12 out of 21). They created the output for a smaller composite and then reused it to generate the remaining occurrences. For example, to create the square pattern, P2 used a modular strategy to reduce the number of repeated actions: *“To go faster, I duplicate the whole group.”*(P2) They found repeated action strategies (4 out of 21) time consuming: *“I have to count the number of copies. This might take some time.”*(P6) And sometimes avoided single action strategies: *“I wanted to do everything at once. I will just do it manually to avoid any problems.”*(P10). Repeated actions provided direct control over the output, but involved tedious manual repetitions. On the other hand, single action strategies reduced repetition, but provided little control over the visual output. Modular strategies were a reasonable trade off between direct control over the output and repetition (Fig. 47).

4.4.6.2 Choosing tools that provide direct access to visual properties

Graphic design professionals use generic GUI manipulations, such as drag and drop and resize handles. They also use application tools that provide direct control over visual properties.

We extract from each strategy the tool or tools used to complete

the visual composition task. We identify two categories of tools extracted from participant strategies: generic GUI manipulations, and application-specific tools (Fig. 48).

Categories of used tools

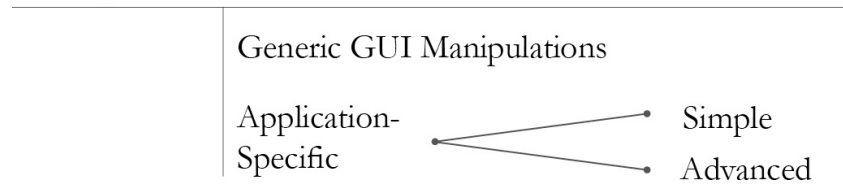


Figure 48: Categories of tools participants used to complete visual composition tasks in the study.

The first category includes generic GUI manipulations such as drag and drop, resize and copy-paste. The second category includes application specific tools. These are tools specific to Adobe Illustrator or to the Adobe Suite. In this category, we identify two types of tools. The first type includes simple application-specific tools. These are strong specific tools (Denning 2001) that act directly on the visual output. They act in a single step and perform one defined action on the current selection. The second type includes advanced application-specific tools. These tools support a complete visual composition task, through changing several parameters. Visually, they are often similar to the Xerox property sheets (Johnson et al. 1989), or the Adobe Pattern tool ².

Participants used 16 different tools in total. They used six different GUI manipulations and 10 different application-specific tools. Participants still chose generic GUI manipulations for tasks with application-specific tools. For these tasks, they used application-specific tools half of the time. For the remaining half, participants used GUI manipulations or combined them with application-specific tools. Fig. 49 illustrates this distribution. Looking at tasks where the application has specific tools, participants used these application tools is almost half of the tasks (yellow in the figure). For the remaining half, participants used either standard GUI manipulations (Green in the figure), or combined these GUI manipulations with application specific tools (Blue in the figure) .

Participants used simple application-specific tools such as the path finder and the pen tool repeatedly in different strategies, for multiple visual composition tasks. Almost all participants(11 out of 12) used the path finder in five different strategies for three different tasks. Ten participants used the path finder to create the logo in Poster B; and seven participants used it to create the logo in poster A.

² <https://helpx.adobe.com/illustrator/using/whats-new-cs6.html>

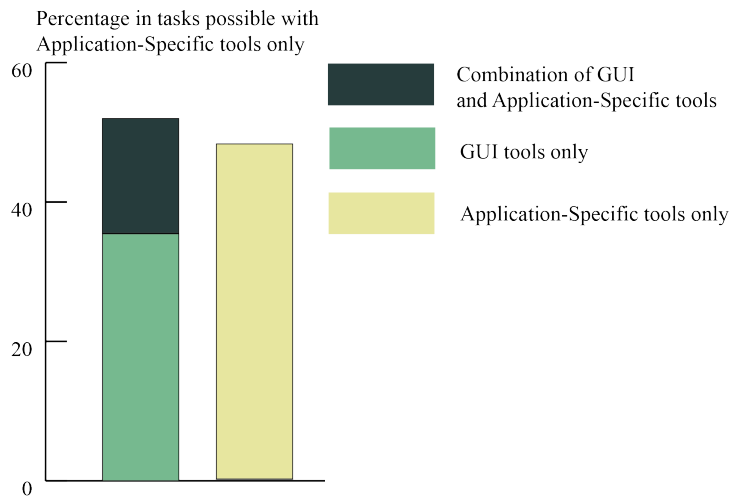


Figure 49: Tool choices for tasks supported with an application specific tool.

On the other hand, very few participants used dialog-box tools, where they manipulate visual properties through sliders, checkboxes, and text fields. Thus, instead of using dialog box tools that generate complete tasks, such as those introduced in later versions of professional design applications, users chose tools that support single actions; and reused them in strategies for different visual composition tasks.

More than half of the participants (7 out of 12) used the pen tool in four different strategies for three different tasks and four participants used the distribution tool in three different strategies for three different tasks. P₄, P₇ and P₁₁ used the tool for two out of three visual composition tasks: vertical and horizontal text box blending in poster A and creating the square pattern in poster B.

Participants rarely used advanced application specific tools. Only one participant (P₁₀) used the pattern option panel ³ to create the square pattern in Poster B; and only two participants (P₉ and P₁₀) used the blend options to create the text box blending in Poster A. Participants who used these tools spent significant time reconfiguring their options. For example, when creating the background text box blending, P₉ spent more than two minutes trying to change the tool options, and was still not sure about the output: *"I feel there are still too many repetitions."* (P₉). These participants wanted to see the results of changing the tool configuration directly on the document.

Graphic design professionals prefer manipulating visual properties with application-specific tools, that do not require additional configuration, and act directly on the object properties. They reuse these tools

³ <https://helpx.adobe.com/illustrator/using/whats-new-cs6.html>

in different strategies for various composition tasks. They rarely manipulate visual properties through option panels similar to the *Xerox Star* property sheets.

4.4.7 *Strategies to manipulate visual properties: Clever Tricks and Bad Hacks*

In the debriefing sessions, we were struck by how often people reflected on their strategies. They were either satisfied with the results and considered their visual composition strategies as new ways of performing the task, or considered them “*a way out*” to finish a task in a very particular context. We call these two types of strategies “*clever tricks*” and “*bad hacks*”. In both cases, users complete the visual composition task. However, users are proud of their clever tricks and would happily reuse them; by contrast, they consider bad hacks cheating and treat them as temporary solutions they would rather avoid in the future. Note that one person’s clever trick may be another’s bad hack. Also note that these definitions are based upon the user’s perception of the software, rather than an inherent property of the software design. We categorized users’ strategies in performing the 12 tasks into clever tricks and bad hacks. We identified 47 unique clever tricks and 43 unique bad hacks.

Clever tricks included cases where the participant first used tools for alignment, proportional positioning or convenient spacing before starting the task. For example, P1 used the ruler to compute the width of the artboard. He then divided this width by the number of repetitions, and used guides to position the object he wanted to replicate within this exact distance. Thus, replication gave a precise result he did not need to fix or verify later. These strategies are clever tricks because they involve planning by computing measures or placing guides. Participants also used tools that provide better visibility and thus ensure better access to the details of the visual elements. For example, while creating a bird shape, P2 created two circles directly in the isolation mode, to ensure that intersection points are properly linked, before moving back to the overprint view for editing the shapes. P5 created a rectangle he placed on top of the logo shapes, just to create an intersection area he would then subtract with the pathfinder tool. Clever tricks occurred when participants used tools that gave them better visibility of the objects and their properties.

Bad hacks usually include fixing or adjusting an acceptable, yet not perfect result. For example, P6 copied each vertical text box of the background text, and moved it manually by pressing the arrow key to achieve the desired distribution. This required keeping in mind the current and remaining number of text boxes as well as the number of

arrow presses throughout the task. P8 created a line that cuts through many squares, and then used the scissors to cut all squares in half. For reproducing the same result on other groups of squares, he preferred duplicating, moving and rotating the squares he already cut: *"It was already too tedious the first time around."* (P8) Participants used bad hacks when they were willing to accept some degree of imperfection. For example, P2 copied the text box several times to recreate the poster background. She then grouped the copies and resized the whole group. Although this deformed the font, she considered the change acceptable. Participants were not satisfied with the results of these strategies because they involve tedious manual manipulations, or because they hide some of the final result properties. These hidden properties require participants to go through extra manipulations for fixing, verifying or undoing unwanted results.

Clever tricks and bad hacks capture users' perception of their visual composition strategies. They also consider the task's context in defining the user intention, and thus, the adaptability of a strategy to the current task.

4.4.7.1 *A recurrent example: Alignment and Spacing*

Five participants created a rectangle to measure the spacing between groups of already aligned objects. Then, they moved this rectangle to copy the spacing among new objects they created. P11 commented on this technique: *"This rectangle is a visual indicator for saving the space between squares."* This was an example of an established clever trick where five different participants used the visual properties of an object for alignment tasks.

The same participant used different alignment clever tricks within the same poster. For example, P2 used four different alignment strategies at different times in the same poster. She created a rectangle for measuring the spacing and reused it five different times. She also copied a group of squares on top of the last row of an existing group of squares to make sure the two groups are aligned. In the same poster, she dragged a text box to the edge of an existing shape to verify the text box alignment. She also created a guide and aligned the logo to this guide. This is an example of the variety of clever tricks, by the same user in the same visual composition task.

Clever tricks and bad hacks describe how participants perceive strategies they chose for a specific visual composition task. We want to see if they also appear in examples of real visual composition tasks and in other applications.

4.5 INTERVIEW STUDY: HOW COMMON ARE CLEVER TRICKS AND BAD HACKS?

We conducted a follow-up interview study with 12 participants, to see if clever tricks and bad hacks still appear in user strategies as they manipulate visual properties, regardless of software application.

4.5.1 *Participants*

We interviewed 12 professionals (5 women, 7 men aged 23-35) from different fields about real projects where they performed visual composition tasks. Professions included: researcher in interactive visualization, software engineer, virtual reality-engineer, front-end developer, product designer, graphic designer and interaction designer.

4.5.2 *Procedure*

Each session lasted around 30 minutes. We asked participants to choose a particular recent or current visual composition they created, such as a book cover, a website, a presentation or a report. We asked them to show us the corresponding artifact, and walk us through the process, step-by-step, of using particular tools and strategies to achieve each result within their chosen application. We probed for examples of strategies that were particularly effective as well as strategies that were tedious or less successful. In each case, we asked them how satisfied they were with their strategies and whether or not they did or would reuse it.

We recorded audio for each interview and took written notes. We also took photos of the final results participants showed us and videos of their interactions with the applications and with the artifacts they presented.

We did not ask specifically for “*clever tricks*” and “*bad hacks*”. We probed for details about participants’ perception of their strategies, their satisfaction and if they reused this strategy for similar tasks.

4.5.3 *Results*

We collected a total of 31 unique examples of clever tricks from all 12 participants. Each participant had between one and five different examples. P7 (Product designer) explained how she could have created a table in Adobe InDesign and then resized the rows to create her book cover. Instead, she created a rectangle and drew lines inside to have more control over the size of the rows and more freedom to modify and reuse it later on. She also reused this same strategy for several covers she showed us within the same book.

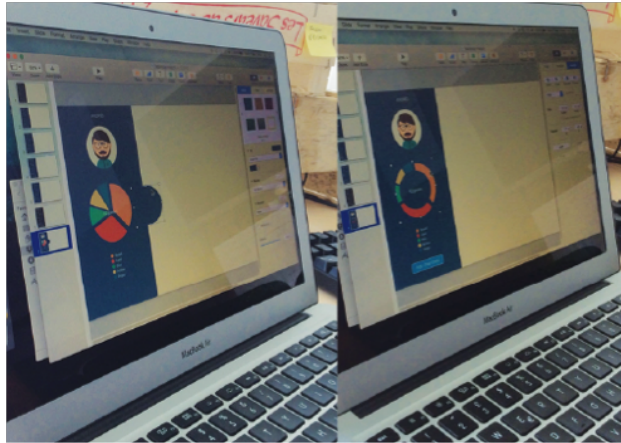


Figure 50: Using a default pie chart, hidden with a circular shape, to simulate menus of an interactive prototype in Keynote.

P9 (Interaction designer) wanted to try different visual effects for the design of an interactive prototype. He used Keynote default charts with random data and then hid the center of the chart with a circular shape instead of creating a ring with different colored stripes (Fig. 50). Similarly, P10 (graphic designer) wanted to create an oven icon in Adobe Illustrator. Instead of drawing a line for the oven handle, she drew a rounded rectangle she then squeezed to appear as a line. This result was closer to her original idea: *“I was more interested in the rounded edges.”* These clever tricks were all strategies participants reused and felt proud to discover and maintain.

Most participants (10 out of 12) admitted using bad hacks. Each participant had between one and four different examples (20 examples in total). P1 (graphic designer) wanted to position characters appearing randomly on the pages of an author’s website. To deal with the characters appearing under the web page content, he changed their transparency in the style sheets. He described this solution as acceptable for the project’s constraints although he wanted to first know how to create safe zones around the characters and have a result similar to his original idea. P10 (Graphic designer) needed to reuse specific drawings from an old version of a website. She rotated the picture, drew a black rectangle to hide the text, and used the pictures on the side: *“This was for a mock-up, I did not use this technique for the actual website.”* P6 (Software engineer) needed to simulate how a tracking system would work in a room. He wanted to create an isometric perspective without using 3D software. He created a 3D grid in Inkscape and set the stroke width of the wall manually to create perspective. He described this as a good enough solution for creating this slide once. These bad hacks were presented as ad-hoc solutions, chosen

only because users could not find a better alternative.

4.6 DISCUSSION

Observing graphic designers uncovered their tool and strategy choices in a design application. Designers choose strategies where they manipulate visual properties in the composition. They prefer strong-specific tools to the ubiquitous property sheets (Johnson et al. 1989). We use these insights to articulate implications for the design of digital visual composition tools.

4.6.1 *Appropriating visual properties*

Graphic design professionals use visual properties such as height, width, color and layering to perform tasks such as distribution, alignment, and shape creation.

More than half of the participants (8 out of 12) used properties of graphical objects for alignment, spacing or distribution tasks. P2 created a rectangle to use its height as a measuring tool to copy the spacing between two squares in poster B. Then, she moved this rectangle to define the exact position for new squares with the same spacing (Fig. 51).

Graphical properties used as tools

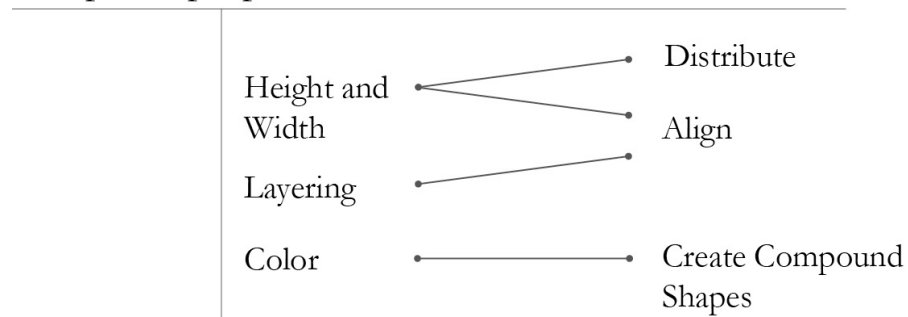


Figure 51: Mapping visual properties and visual composition tasks they were used to complete

For text highlighting, all participants created colored rectangles they sent to the background. For this task, we expected users to look for a highlighter they would grab and apply to the text lines, similar to the tool available in other applications. Yet, nine out of the 12 participants searched for a tool to change the character background color. For example, P1 and P2 first looked in the character tools. P11

and P12 wanted to change the underline stroke thickness and color to highlight text. While these attempts were not successful, they revealed users' need to access and manipulate visual properties and use them as tools for visual composition tasks.

Designers appropriate visual properties when they can access them directly on the interface objects. In these cases, they changed visual properties not to edit their values for graphical objects, but to use these properties for visual composition tasks. For example, a recurrent strategy consists of creating a rectangular shape and applying the background color to this shape. The user then places this rectangle on top of any graphical elements they want to hide. The manipulation of height, width, and color is not the user's goal. Instead, these properties serve to hide specific objects from the composition.

4.6.2 *Implications for design*

Our observations of how designers manipulate visual properties in an application suggest the following directions for designing tools for manipulating visual properties:

- Direct access to visual properties:
Participants use strong-specific tools more often because they act directly on the visual output. Participants use these tools instead of property sheets (Johnson et al. 1989).

Property Sheets appeared in the xerox star (Johnson et al. 1989), but also in the latest versions of visual composition tools such as Adobe Illustrator CC ⁴. Panels in Adobe Illustrator provide access to visual properties such as size, height and width through text fields and check-boxes.

The initial Xerox property sheets are based on the principle of progressive disclosure: details should be hidden from users until they ask or need them (Johnson et al. 1989). In this perspective, visual properties are part of the intrinsic properties of an object. Visual properties are details hidden, within property sheets. They are displayed on demand and not in the same dialog box. An "Other" button reveals more visual properties upon request.

Progressive disclosure of visual properties is adapted to typing text or performing calculations. However, visual composition focuses on experimenting with visual properties such as height,

⁴ <https://helpx.adobe.com/illustrator/how-to/live-shapes.html?set=illustrator--whats-new>

size, color, position and layering. The user's main goal is to assign values for these visual properties to build a visual concept. For this reason, hiding visual properties in dialog boxes like the property sheets is not adapted to designers' visual composition practices. In fact, graphic designers in our study did not often choose this type of tools. They preferred tools that provide direct access to visual properties. Thus, visual composition tools should support direct access to visual properties. Visual properties should not be details linked to graphical objects, displayed on demand. They should be independent objects users can directly access.

- Independent representations of visual properties:
Current graphical user interfaces provide different representations of the same visual property, depending on the current selection. For example, to change the stroke and fill colors in Adobe Illustrator, users need to select the stroke or fill box from the tool panel. If they select the stroke box, they get the representation in (Fig. 52a). If they select the fill box, they get a different color representation (Fig. 52b). These two standard color representations also suggest different interaction techniques, and provide access to different color parameters. The stroke representation has hues only. The fill representation resembles the standard color picker. The stroke representation provides an eyedropper tool to select a hue from the spectrum. The fill representation provides a cursor to select saturation and brightness, and another cursor for hue. It also has text fields for entering specific HSB, RGB, or Hex color codes.

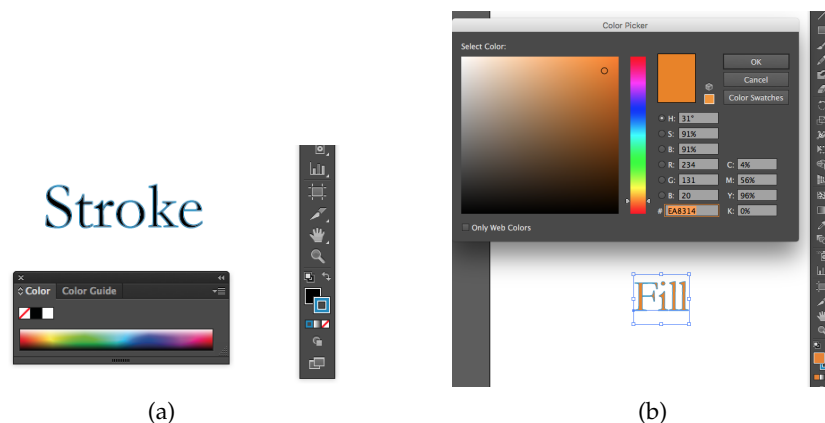


Figure 52: Two different representations to access color in Adobe Illustrator CS6.

Visual composition tools should provide representations of visual properties, independently of the current selection. These

representations should also provide consistent interaction techniques.

4.7 SUMMARY

Graphic designers manipulate visual properties directly in the composition. They choose tools within the application, that provide direct access to these properties. Graphic designers find novel ways to use visual properties for composition tasks. Users perceive these strategies as *Clever tricks* they create and maintain, or *Bad hacks* they use in specific situations.

Visual composition tools should provide direct access to visual properties. They should provide access to these properties, independently of the current selection.

4.8 CONTRIBUTIONS

1. Creation of two isomorphic visual composition tasks with Panagiota Tziova.
2. Found that graphic designers manipulate visual properties in the composition. They choose tools, from the application, that provide direct access to visual properties.
3. Found that graphic designers create visual composition strategies based on finding new ways of using visual properties. They perceive these strategies as *Clever tricks* they maintain, or *Bad hacks* they use for specific visual composition tasks.
4. Implications for the design of tools for manipulating visual properties for visual composition tasks.

ARTISTS AND DESIGNERS COLOR MANIPULATION PRACTICES

Visual composition tools include color tools, such as the color picker, to support selecting a color from a color representation. This chapter investigates how these tools support artists' and designers' color manipulation practices. It presents an interview study with artists and designers about their color manipulation practices. We found that artists and designers create and use personal instead of generic color representations. Artists and designers manipulate the visual properties of color, such as size, position and layering, in the context of the surrounding visual properties. They also combine color with other visual properties such as texture. Finally, this chapter suggests implications for the design of color manipulation tools.

5.1 INTRODUCTION

Bauersfeld and Slater highlight problems with current color tools, including the traditional color picker (Bauersfeld and Slater 1991). Color tools usually hide the objects to which color will be applied. They also use the gamut of the color model as an interface to the tool, which does not always support the color manipulation task at hand.

The ubiquitous color picker usually includes three common features: a visual representation of a specified color model, the organization of displayable colors into a three-dimensional color space, and controls to change parameter values within that space (Douglas and Kirkpatrick 1999). Users can select individual colors from the color space, either with the mouse or by specifying a three-digit code, such as an RGB value. Some color pickers also allow users to select a color from a pixel in an image or from existing color swatches.

Despite being ubiquitous, color pickers have changed little over the past 25 years. Fig. 53 shows almost identical layouts and controls for three common color pickers; the only new features are their underlying color spaces, which have been updated according to research in color perception (Faul and Ekroll 2002) and representation (Meier 1988).

Manipulating visual properties such as color, depends on the existing tools and representations (section 2.5.1). For this reason, the tools visual composition tools provide to support color manipulation tasks affect how artists and designers manipulate color. What tools

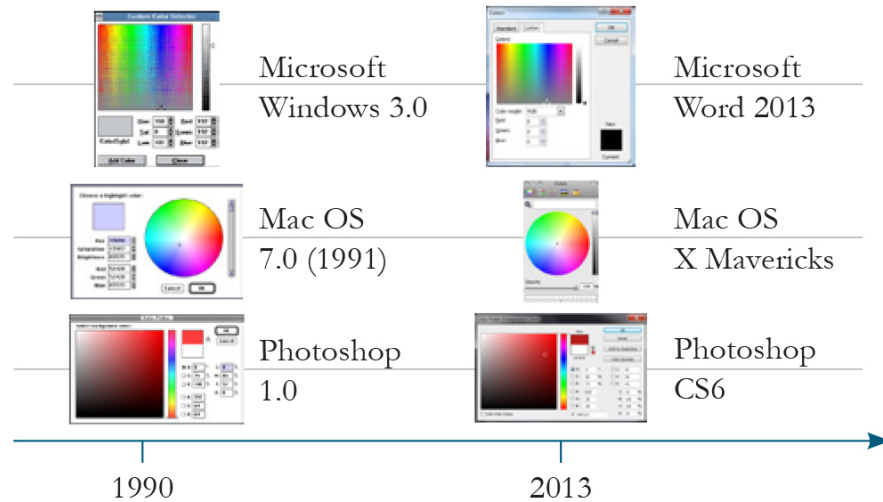


Figure 53: Today's color pickers have changed little since 1990.

do artists and designers use to manipulate color? Are existing color tools enough to support their practices?

5.2 CONTEXT

Chapter 2 (section 2.5.1) describes color representations popular among artists and designers as they manipulate color. Most current digital color tools, including the color picker, still focus on creating representations of these color models.

Our goal is to improve how authors of digital media manipulate color to achieve desired effects. This requires a more detailed understanding of current color practices, beyond the use of the ubiquitous, but unchanging color tool: the color picker. We are interested in the tools and techniques artists and designers use to incorporate color into their work, especially how they produce specific color effects. Although digital color tools are our primary concern, we also want to understand physical color practices that may inspire new ways to manipulate color online. We run an interview study to investigate artists' and designers' color practices as they create their artifacts.

5.3 PARTICIPANTS

We observed and interviewed eight artists and designers (4 men, 4 women; age 23-45) who consider color an essential part of their work practice (Fig. 54). Their professions included painter, illustrator, ce-

ramist, spatial designer, graphic designer, product designer, service designer and interaction designer.



Figure 54: Eight artists and designers demonstrated how they manipulate color to achieve effects in both physical and digital media.

5.4 PROCEDURE

We interviewed participants in their studio or office for about one hour. We asked participants to tell stories about their use of color in recent projects and to show us the resulting artifacts. We recorded audio for each interview and took written notes. We also recorded video of participants' interactions with the objects they created, and photographed each artifact and any related color creation or manipulation tools. We probed for situations in which their interaction with color was particularly effective, but also when it was extremely difficult or impossible. We framed the interviews around a core set of questions (Appendix B) that evolved throughout the discussion.

We conducted *critical object interviews* (section 1.3.1) grounded in participants' color manipulation practices. We asked participants to show us recent artifacts or projects they created. We asked for details about the specific steps participants went through to select and apply color.

For each artifact, we asked the participant to explain how she obtained: *this color*, meaning the color we see on the artifact. Starting from the final object, we probed for the specific steps participants went through to find and apply colors. We also asked about the on-line and physical tools they used, and any intermediary representation before applying color to the artifact.

5.5 DATA COLLECTION

We created a visual representation for each critical object story (Fig. 55). We call these representations *Color Portraits* (Jalal, Maudet, and Mackay 2015); they capture the situated nature of the story-based data we col-



Figure 55: Color Portrait illustrating a Critical Object Story.

lected from the interviews.

Color Portraits represent the components of a critical object story in a single-page format. We use Color Portraits to compare and classify critical object stories. We later showed Color Portraits to the participants to verify the details of the stories extracted from their interviews.

5.6 DATA ANALYSIS

We used Grounded theory (GT) (Corbin and Strauss 2007) (section 1.3.2) to uncover artists' and designers' color manipulation practices in their recent projects. We extracted color manipulation categories from crit-



Figure 56: Color Portraits help comparing participants' color manipulation stories.

ical object stories. In the initial phase, we went through all the stories and extracted color manipulation codes, then categories (Fig. 56).

5.7 RESULTS AND DISCUSSION

Participants explained the strategies they followed to create specific color effects. Artists and designers prefer personal to generic color representations. They manipulate visual properties of color, such as size, position and layering. They also combine color with other visual properties, such as texture and light reflection. Their color manipulation depends on the surrounding visual properties in the composition.

5.7.1 *Personal color representations*

Most color tools, including the color picker, provide color representations that depend on the color model. These representations are not easy to adapt to the current constraints of the artifact and to the user's concept. Participants create different types of representations they use to select their color schemes. They also use these representations to communicate meaningful information about their color manipulation process, or to reveal how an activity evolves over time.

5.7.1.1 Sampling and tweaking colors

Artists and designers derive their colors from different sources, either from the color space provided by a color picker or from an existing sample. Although users can apply these colors directly, they more often tweak them, by modifying them before use in their artifact. Fig. 57 defines two activities related to color selection. Artists and designers select their artifact colors from previous color samples. When working on a new artifact, they select their first colors from personal representations. They then modify these colors before using them in their artifacts.

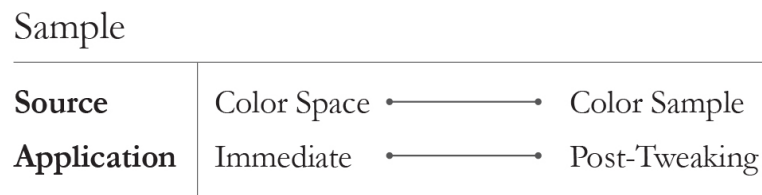


Figure 57: Users pick colors from diverse sources, and often tweak them later.

Most participants (7/8) used personal representations, including websites, color palettes and photographs, as well as physical objects such as ceramics or textiles. P4 (Graphic designer) chose a blue from a catalog to ensure that the printed version of his poster would appear exactly as he wanted. P7 (Product Designer) selected his colors from a book he likes to read. He searched online for the digital book cover, and used the eyedropper tool in InDesign to sample the cover's colors. P1 (Painter) described a project where he mixed his paint colors based on printed frames from a video he had created. He identified video frames with the colors he wanted. Then, he printed these frames and taped the printed papers on his easel. Finally, he visually compared the results of his color mixtures to the printed frames while painting.

Few participants (2/8) still used generic color representations to select their starting colors. P6 (Product Designer) used Adobe Photoshop's color picker to recreate a particular blue she had already used for a website, and P4 (Graphic designer) tried to create "unusual colors" using Adobe Illustrator's color picker: "Salmon colors are strange colors, they are at the border of ugly, but I like to play with ugly colors, as long as they fit with the concept" (P4).

Even with personal representations, most participants (7/8) still tweaked the colors before using them in their final artifacts. They started with a sample and then tweaked it before applying it to the

artifact. For example, P2 (Exhibit designer) sampled colors from paintings she found online, and then used the color picker to change the samples before adding them to her mood board. P7 (Service designer) created a palette using Adobe Color. She extracted colors from the screen with InDesign's eyedropper, changed one of them, and then used the resulting palette in her final design. P1 (Painter) started from a sample of three tubes of oil colors and mixed them to obtain the final colors for his painting (Fig. 55).

Artists and designers prefer personal representations to generic color spaces. They search for their colors in previous projects, in online or in physical objects. They use these personal representations because they contain information relevant to their current artifacts. Artists and designers tweak colors before using them in their artifacts. Current color tools provide generic color representations participants use once they know the color they want to use. Prior to this selection, they create their own, personal representations, outside the tool.

5.7.1.2 *Creating personal informative color representations*

Artists and designers sometimes observe color changes that reveal useful information about interim states in their work process. They use color to visualize the evolution of their work over time. Fig. 58 defines two activities related to the color manipulation process, including: using colors that result from other activities and revealing ongoing processes. Artists and designers also create informative color representations of their color manipulation process. They save the source, the destination, and the interim colors they explored before selecting their final color scheme. Fig. 59 defines two activities related to creating informative representations of the color manipulation process. They include: preserving source materials and final artifacts, and capturing intermediate steps in the color manipulation process.

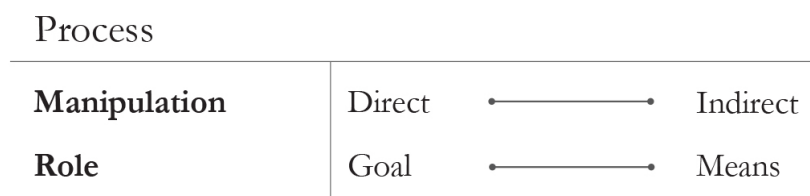


Figure 58: Artists and designers use color representations to reveal useful information about interim states in their work process.

Artists and designers use personal color representations to reveal how they created an artifact, or the amount of time they spent on

its creation. These color representations provide additional information about the evolution of an activity over time. Half the participants used color to indicate how they created an artifact or the amount of time spent on its creation. P6 (Product Designer) observed colors to determine important details about her design process: *“Just by looking at the pot, you can see how many layers I created ”*. In another project, she heated metal chairs, which caused the metal to change colors. She stopped the process when she liked the color and applied a coating to stabilize the color.

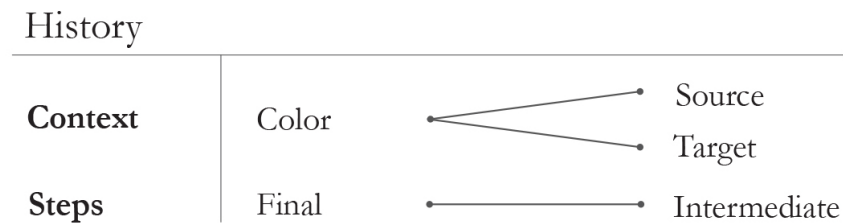


Figure 59: Users record color sources and targets, as well as intermediate steps

Artists and designers use personal color representations to capture their color manipulation process. Half the participants (4/8) sought ways to save meaningful intermediate steps in the process of creating a final color. For example, P5 (Ceramist) kept samples of every color she created over the past decade, as well as notebooks containing personal names, codes and the numbers of trials needed to obtain each color. A few participants (2/8) also kept track of source colors. For example, P7 (Service designer) saved images he downloaded from the Internet: *“I use these images to extract colors for my palettes and I keep them for later reuse.”*[P7]. Other participants saved their final palettes with the resulting artifact. For example, P4 (Graphic designer) placed different-sized rectangles with each final color into the unused space beyond the margins, and saved them as part of the final document. Some participants (2/8) wanted to return to a previous use context, with both the initial color source and the final artifact. For example, P5 (Ceramist) used several previously created red tiles to develop a nuanced set of three slightly different red tiles for another client.

Current color tools provide little support for histories. Apart from saving previously used colors, usually outside the document, color tools do not support creating informative representations of color changes over time. Artists and designers are aware of how color changes throughout their work. In fact, these color changes provide additional information about their work processes. They want color representations that reveal these changes. They also want color rep-

representations that capture the meaningful intermediate steps in their color manipulation process.

5.7.2 *Manipulating color in context*

Artists and designers experiment with colors. They develop rich color manipulations, according to the surrounding visual properties.

5.7.2.1 *Manipulating visual properties of colors in context*

Artists and designers create coherent color sets and manipulate them in groups. They create color palettes: representations of color schemes they use in their artifacts.

Palette		
Size	Fixed	Resizable
Layout	Juxtaposed	Layered
Position	Static	Dynamic
Manipulation	Individual	Group
Context	Independent	Dependent

Figure 60: Artists and designers manipulate the visual properties of color swatches within a palette.

Artists and designers manipulate color palettes by manipulating the properties of individual color swatches. They change the position, size, and layering of these swatches.

Most participants (7/8) created related sets of colors rather than separate individual colors. P2 (Exhibit designer) took photographs of related colored objects. For her, “each picture is a different palette”[P2]. Although the pictures contained the same objects, their positions differed, which resulted in different color compositions (Fig. 61).

Artists and designers also manipulate all the palette colors simultaneously. P8 (Illustrator) created one palette and then modified the hue of each color by the same amount, generating a new palette. Simultaneously adjusting one property for the entire set of colors allowed her to maintain a related, harmonious color palette.

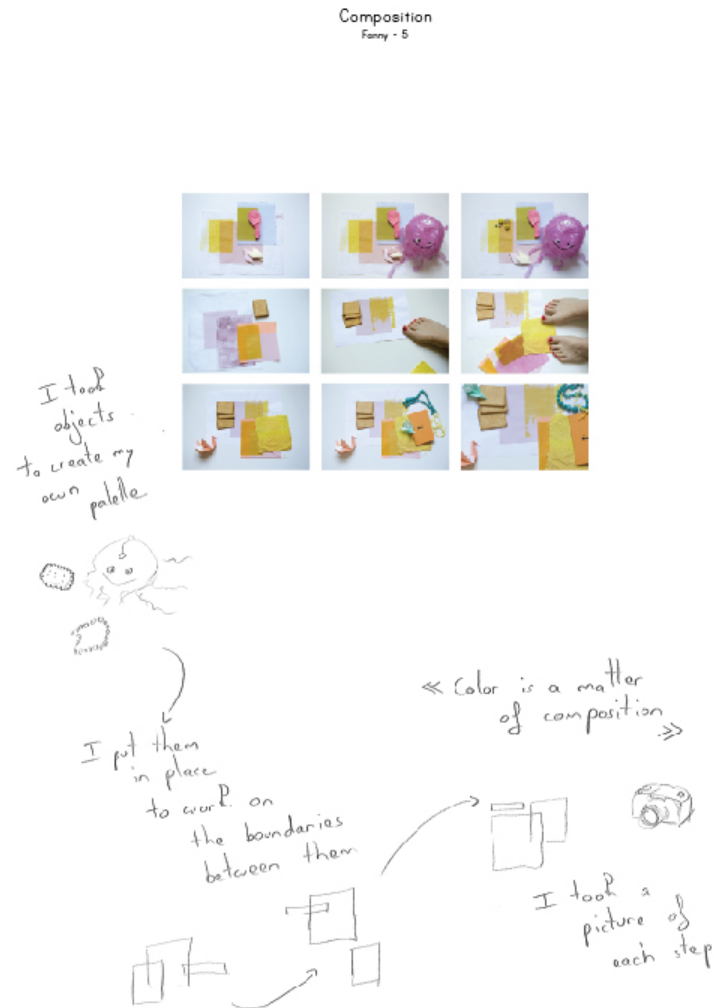


Figure 61: Changing sizes and positions of colored objects creates different palettes.

Color pickers provide a list of past color choices, but with no context. Color swatches are usually juxtaposed, of the same size, and in fixed positions. Artists and designers manipulate colors in context. For this reason, they manipulate the visual properties of individual colors in a palette, or those of the whole palette, to create coherent color compositions. Color visual properties they manipulate include absolute and relative size, position, and layering of individual swatches (Fig. 60).

5.7.2.2 Combining color with other visual properties

Colors are affected both by their surrounding colors and by other adjacent visual properties. (Fig. 62) defines two activities related to color composites, including: composing and decomposing multiple components, and manipulating these components individually or together.

Artists and designers combine color with other elements, such as

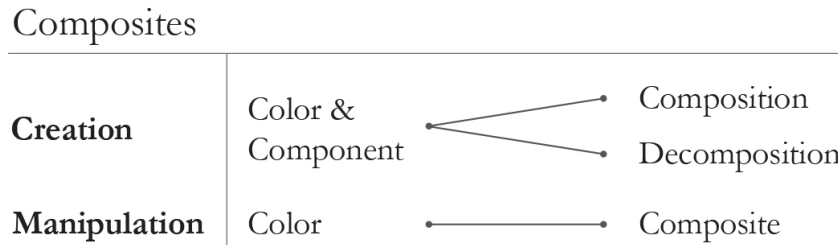


Figure 62: Artists and designers create composites, by combining color and other visual properties. They manipulate a composite as a single element. They also decompose it to extract composing visual properties.

texture and light reflection. They manipulate the resulting combination as a single visual element. Half of the participants (4/8) had examples of combining color and texture. P2 (Exhibit designer) described the screen as “*a flat surface that does not always transpose the richness of the physical world*” (P2). To reproduce the yellow texture of a tablecloth on the screen, she scanned it and used the resulting image to provide the effect she wanted. Color and texture, are equally important in her process (Fig. 63). P8 (Illustrator) used Photoshop to manipulate a color and a texture she paired together in several illustrations. She used this combination as a single visual element, but had to manipulate color and texture separately each time, and later combine them. P1 (Painter) created a special preparation that added a particular type of light reflection to each color. He considers this combination of color-plus-reflection as his personal signature. He would apply this preparation to his colors and think of the resulting colors only when combined with this preparation (Fig. 55).

Most color tools propose separate color, and pattern or texture palettes. Users need to manipulate these elements separately, and combine them later. Artists and designers do not only manipulate color in the context of other visual properties. They also manipulate the combination of color and other visual properties, such as texture or light reflection. They often mention the “effect”, or the “result”, which is the visual output of combining color and another visual property into a single unit for manipulation.

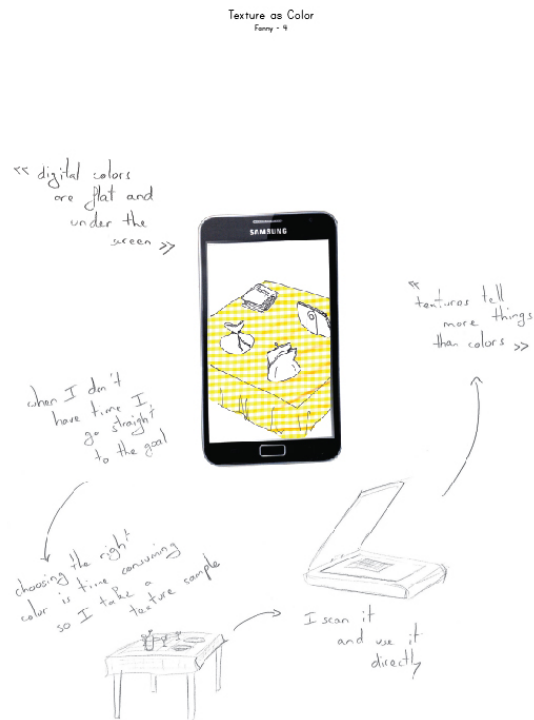


Figure 63: P2 combined color and texture into a single visual unit for manipulation.

5.7.3 Manipulating personal color representations in context

Most color tools provide generic representations, based on digital color models (RGB, CMYK, HSV...). Representations associated with these color models are generic. They do not contain personal references to help users select one or more colors for the specific artifacts they want to create. For this reason, when they need to select colors, users often start using these tools once they already know the colors they are looking for. Tools such as the color picker, support selecting a color from an existing representation. They do not help users “choose” their colors. These representations do not adapt to users’ media, artifacts, and to the meaning users attach to each color.

Color manipulation practices appear in user practices in two main forms. First, users want to focus on color by manipulating its visual properties- such as size, position and layering- in the context of other colors. Second, they want to create “composites”, by combining color with other visual properties, and manipulating the combination as a

single visual element. These two directions reveal the changing meaning color acquires depending on its visual properties, and on its surrounding visual elements in the composition. Thus, supporting color manipulation practices requires balancing the focus between color and its context of use.

5.7.4 *Implications for design*

Our observations of how artists and designers interact with color suggest the following directions for supporting color manipulation. Visual composition tools should support:

- Creating personal color representations. Users should be able to adapt color representations to their media, to their visual concepts, and to their work processes. Color tools should support creating color representations that capture the meaningful steps in the color manipulation process. They should also support creating informative color representations, that reveal how a process evolves over time.
- Manipulating the visual properties of color such as the size, position, and layering of different colors in a palette. This should be in the context of other colors, and of other visual elements in the composition such as text, shapes, and images.
- Combining color with other surrounding visual properties. Color tools should support creating new visual elements based on this combination.

5.8 SUMMARY

Artists and designers create personal color representations. They care about the information a color representation contains (references to existing artifacts, particular themes or ambiances...). For this reason, they want to create color representations that capture their color manipulation process, and that reveal how their activities evolve over time.

As they manipulate color, artists and designers consider surrounding visual properties. These visual properties include size, position, and layering. Artists and designers manipulate these properties to create specific color changes. They also combine color with other visual properties in the composition.

5.9 CONTRIBUTIONS

1. Method for extracting specific story elements from critical object interviews, and creating a representation of critical object stories: *StoryPortraits* with Nolwenn Maudet.
2. Found that artists and designers create and use personal, informative color representations. They manipulate the visual properties of color, in the context of surrounding visual properties. They also combine color with other visual properties, and manipulate the combination.
3. Implications for the design of color manipulation tools. Visual composition tools should support creating personal color representations. They should allow manipulating the visual properties of color, in context.

GRAPHIC DESIGNERS' LAYOUT PRACTICES

Graphic designers use tools, such as the grid, to specify values for spatial properties. We want to know the visual properties professional graphic designers use to structure content, and the tools they use to create their layouts. This chapter presents the results of 12 interviews with graphic design professionals about their layout practices. Graphic designers use spatial properties such as repetition, symmetry, size and position to activate their composition space. The traditional grid allows fixing values for these properties according to the constraints of the page format. As they create their compositions, graphic designers want to create layouts that evolve in space, throughout the components of a composition, and in time, across related compositions. Graphic designers define fixed and flexible properties. They manipulate the links between these properties throughout their composition, and define rules for how these properties change over time.

6.1 INTRODUCTION

The grid is designed for organizing print content, where the graphic designer knows in advance all the properties of the final design, including content length, page size, binding, etc.

Traditional visual composition tools base their structuring features on grids, guides, rulers, and masters. These tools embed the assumptions of a static and fixed output. How do graphic designers organize content? What are the visual properties graphic designers manipulate when they organize content? How do graphic designers manipulate visual properties when they create their structures?

6.2 CONTEXT

Chapter (section [2.5.3](#)) presents visual composition rules that guide graphic designers when structuring content. In this process, graphic designers activate compositional factors such as size, position, repetition and orientation. These compositional factors help the designer establish a hierarchy among their content elements. Layout principles guide graphic designers in structuring content. These principles do not define how the final composition looks. Designers choose tools and define the behavior and hierarchy of visual elements to create their layout.

We are interested in graphic designers' layout creation practices. We

want to know the tools and techniques graphic designers use to create their layout both for digital and print media.

Our goal is to improve how authors of digital media manipulate properties to create their structures. This requires a more detailed understanding of current layout practices, beyond the use of the expected spatial properties such as size and position, traditionally represented in the grid (section 2.5.3).

We run an interview study to investigate graphic designers' layout creation practices as they organize content.

6.3 PARTICIPANTS

We interviewed 12 graphic designers (5 male, 7 female), age 24-50, with 4-25 years of experience (mean=10,5) who work in various environments (freelance, studio, agency) and create layout for digital media (2), print media (2) or both (8).

6.4 PROCEDURE

We interviewed participants in their studio or office for about two hours. We asked participants to tell stories about their structuring practices in recent projects and to show us the resulting artifacts. We recorded audio for each interview and took written notes. We also recorded video of participants' interactions with the objects they created, and photographed each artifact and any related structure creation or manipulation tools. We probed for situations where their interaction with layout was particularly effective, but also when it was extremely difficult or impossible. We framed the interviews around a core set of questions (Appendix B) that evolved throughout the discussion.

We conducted *critical object interviews* (section 1.3.1) grounded in participants' layout practices. For each artifact, we asked the participant to explain how they structured the content in this artifact. Starting from the final object, we probed for specific steps participants went through to create their layout. We were also interested in the online and physical tools they used, and in any intermediary formats in which they represented or manipulated layout before applying it to the artifact.

6.4.1 Data Collection

We created a visual representation for each critical object story. We call these representations *Layout Portraits* (Jalal, Maudet, and Mackay 2015); they capture the situated nature of the story-based data we col-

lected from the interviews (section 1.3.1).

Layout Portraits visualize the components of a critical object story in a single page format. We used Layout Portraits to compare and classify critical object stories. We later showed Layout Portraits to the participants to verify the story details.

6.4.2 Data Analysis

We used grounded theory (section 1.3.2) to uncover designers' layout manipulation practices in their recent projects. Appendix B provides additional details about the categories extracted from participants' stories using grounded theory.

6.5 RESULTS

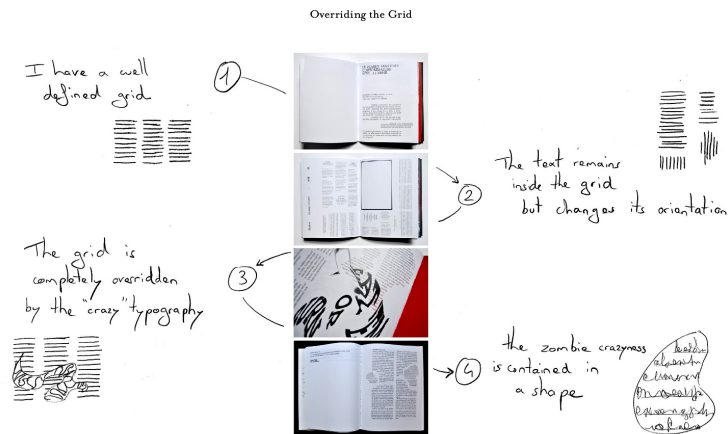
6.5.1 Manipulating links between spatial properties

Graphic design professionals establish links between the spatial properties of content elements. Participants use visual weight, by manipulating the relative proportions of content over white space. They create borders by placing lines, guides or a grid around the content. Participants manipulate other spatial properties such as position and layering. They are interested in the relative positions of elements on the page.

Eight participants used visual weight and size to create their structure. P7 first explored and played with relative weights of visual elements: *"the content creates visual masses I use to establish the concept of my book structure"*. Similarly, P8 created a dynamic column system and adapted it for all of his website layouts. He called it the *"grosso-modo grid"* because it worked using approximate proportions: *"tiny, little, big and huge"*. He used this dynamic column system to link visual properties such as size and position. He then changed this link to adapt to different website layouts.

Participants used *"borders"*, *"guides"* and *"grids"* to create their structures. P1 defined her website structure using InDesign guides. Some of these guides defined the margins while others marked elements' position and alignment. When she finished the first page, she duplicated the entire file to reuse her guides in other pages. P5 established a well-defined grid in the beginning. She then changed text orientation, font size and type, while remaining within the initial grid: *"The text remains inside the grid, but I change its orientation"*. (Fig. 64)

Participants created links between elements' positions to create their



EB.5

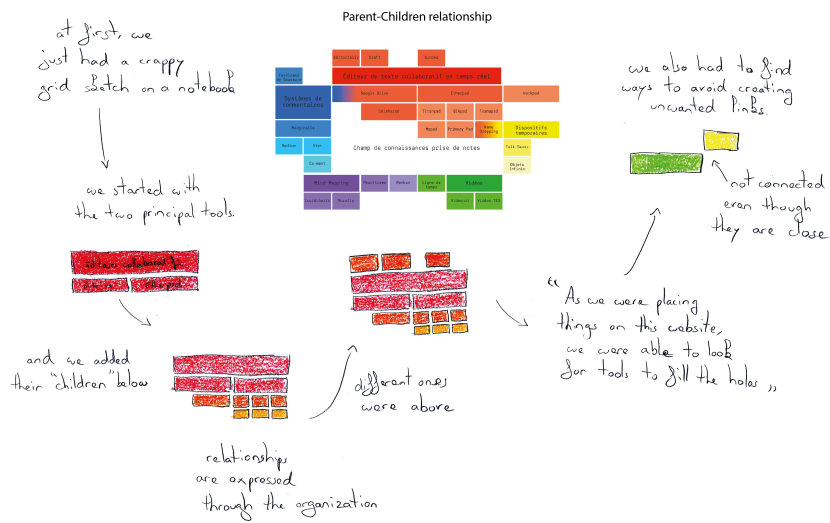
Figure 64: P5 changed text orientation, while remaining within the grid.

structure. P1 created an initial layout for four master pages of a website. She defined the relative position of visual elements on these pages: "All master pages should have the same look. This is a global positioning". She started by drawing and positioning elements on paper before creating the layout in Adobe Photoshop. P9 defined the precise position of a recurrent caption that appears on all the pages of his book: "This way, the reader will not feel lost if we move other images around".

Participants also created links between elements' layerings. P4 designed a unique book layout by superimposing several layers. Each page contains three layers, one for each primary color used to print black in the CMYK model. By cutting specific areas in each layer, she revealed colors from the two other layers. In this example, P4 manipulated the order of her layers to create her book structure.

Participants created links between the properties of content elements based on the content they wanted to structure. Ten participants used content properties to structure their content. P4 assigned a number of occurrences and a different color to each content category. For example, a title would be repeated five times, but a subtitle only three times. She used repetition and color to emphasize specific content elements depending on their role in the composition. Participants also used semantic relationships between content elements to create their structure. P11 created a tree structure to layout a webpage of text-editing tools. She started with the main tools, using lower tree nodes

to place more recent tools based on the main ones on top (Fig. 65).



56.1

Figure 65: P11 used semantic properties of the content to create a link between visual properties.

Participants created links between the visual properties of content elements based on the properties of their destination format. Ten participants used the properties of their destination to create their structuring concept. For example, P5 created a book with sheets folded in two, nested, and stapled. She used physical properties proper to the binding process to structure her book. Images spanned full sheets of paper. Once folded, the left part of the image is juxtaposed with the right part of another image, “*creating an interesting confrontation*” (P5).

Graphic design professionals establish, maintain and modify links between spatial properties such as size, position and layering. They create these links based on the source: the content they want to structure, or the destination: the format where their content will appear. Graphic designers use these links to create and reuse their structuring concept throughout the composition.

6.5.2 Flexible and fixed visual properties

Graphic designers create consecutive layouts that need to be coherent (pages of a book, issues of a magazines). They want to create layouts that evolve over time. For this reason, graphic designers select fixed, and flexible visual properties. Flexible properties ensure variety, while fixed properties create unity across several related com-

positions.

Participants chose fixed values for specific visual properties. They used these fixed properties as creative constraints in their composition. Participants defined other properties as flexible, and changed their values across composition instances. Ten participants used flexible properties in their structuring concepts. P6 used the same logo for different issues of the same poster. He defined some visual properties of the logo as fixed. For example, the logo's size and position in the poster remained unchanged. He manually reused the previous version and slightly modified some of its properties, such as color. In this example, P6 created an evolving structure, where all poster logos have a set of similar visual properties, while each logo still holds a unique identity (Fig. 66).

P10 created the layout for four different issues of the same magazine.

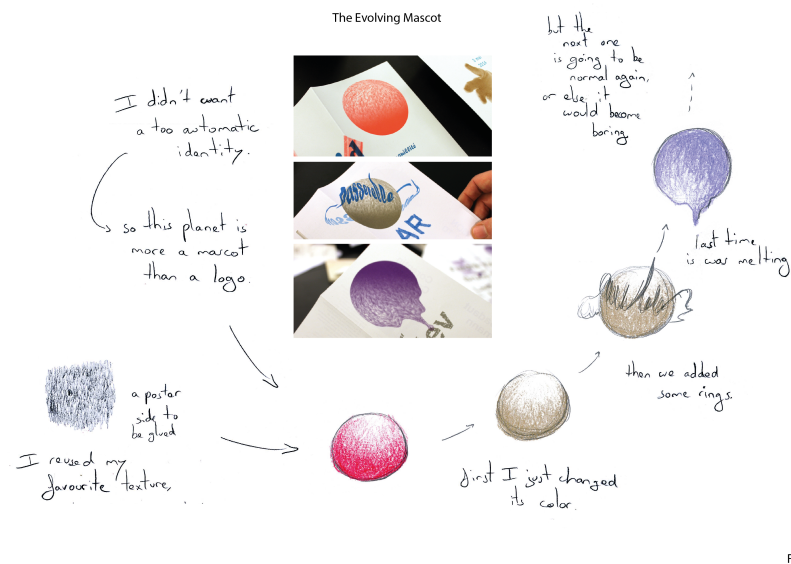


Figure 66: P6 defined fixed and flexible visual properties while structuring visual elements within a poster.

In the first issue, he chose a classical grid, where most of the visual properties are fixed. In each issue, he made new visual properties flexible and changed their values: *“as time goes, I took more freedom with the grid”*. P11 wanted all her website content to fit on a single page. Thus, she fixed the page size and varied the layering of her content elements.

Graphic designers define fixed and flexible visual properties as they structure content. They want to identify and reuse fixed prop-

erties. Flexible visual properties, on the other hand, vary within the same composition, or across different related compositions.

6.5.3 *Guidelines for dynamically changing visual properties*

Graphic designers create rules that define how visual properties change over time. Participants manually defined how visual properties change throughout the composition. They created rhythms and patterns through alternating specific values for the same visual property. For example, P7 created a two or three columns grid-system based for a cooking book. She applied one of the two grids for each page: *“I created a rhythm based on the modular repetition of this system”*. This temporal pattern guided the reader through the content.

Participants manually created and applied these dynamic changes in the values of visual properties. They established a temporal pattern, articulated it during the interview, but could not embed it in their tools.

Participants also wrote programs to define rules for dynamic visual properties. Half the participants used code to articulate these rules in the form of functions. They wanted to obtain predictable, yet dynamic changes for specific visual properties. P3 defined parameters for a generative algorithm to create images with high graphical diversity. In each computer-generated image, visual properties change based on the algorithm. In this project, P5 wanted to find *“the shortest functions that produce the greatest graphical diversity”*. P9 created a dynamic scrolling principle by defining the direction and orientation of scrolling arrows for a website. The specific content to display when scrolling is interactively defined as the user navigates through the website (Fig. 67).

Graphic designers define rules and constraints for dynamically changing visual properties. These dynamic changes contribute into building their structuring concepts. Graphic designers define these rules at once, using program functions. They have to apply them manually, and repeatedly throughout the composition when they use traditional desktop publishing tools.

6.5.4 *Changing the digital nature of visual elements to access specific visual properties*

One surprising practice consists of changing the nature of visual elements in an online document. When they wanted to access specific visual properties, graphic designers changed the nature of some visual elements, the application, and the tools they used for some compositions. P4 wanted to manipulate typography as *“form”* (section 2.5.2).

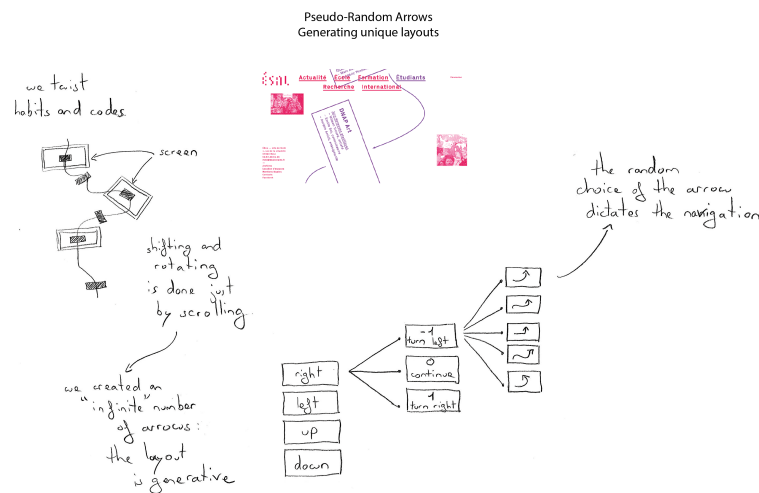


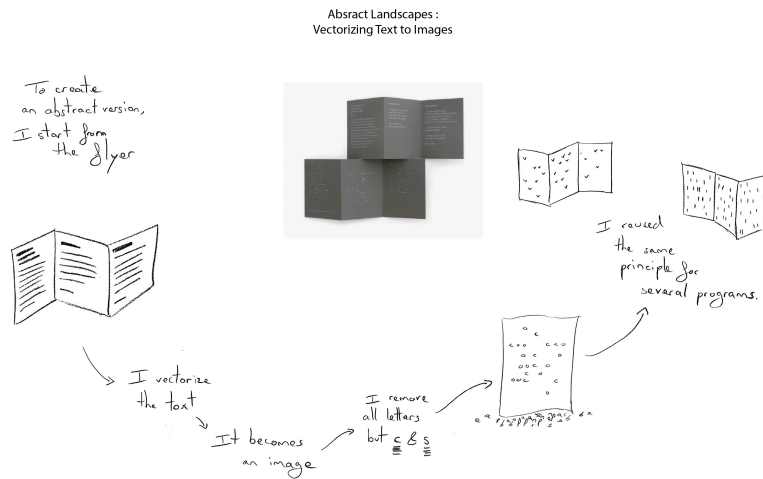
Figure 67: P9 programmatically defined the scrolling behavior for navigating a web page.

She wanted to consider each individual letter as an image to manipulate its size and position individually. For this reason, she converted the text document into an image. She opened this image in Photoshop, and kept only two letters from the original text to structure a set of flyers. She manipulated the position of each letter to create unique shapes for each flyer (Fig. 68).

Graphic designers create their structuring concepts as they define the behavior of visual properties. Visual composition tools do not always provide access to all the visual properties of graphical objects. In these cases, graphic designers need to change the visual nature of graphical objects, the application and the tools they use to manipulate these properties.

6.6 DISCUSSION

Interviews with graphic designers revealed that visual organization practices do not focus only on fixed values for spatial properties, captured in the grid. These practices include creating links between spatial properties. They also include establishing rules for flexible visual properties, that take different values, as the composition evolves. We use these insights to articulate implications for the design of digital layout tools to support layout creation practices.



FM.2

Figure 68: P4 transform text to an image to control the position of each letter individually.

6.6.1 Beyond composition factors: flexible visual properties for dynamic layouts

Graphic design guidelines urge designers to use spatial properties such as position, size, and alignment, to create a hierarchy of visual elements (section 2.5.3). For these tasks, the traditional grid helps create a fixed layout, based on the format constraints of the destination page.

In their practices, graphic designers go beyond assigning fixed values to spatial properties. They create links between these properties, and manipulate these links throughout the composition. They define fixed, but also flexible properties that change value within the composition, or across related compositions. Graphic designers also define rules for how these properties evolve over time.

The traditional grid (section 2.5.3) does not support the representation of links between spatial properties. Graphic designers still create these links manually. They have to recreate instances of their structure for individual components of their composition.

6.6.2 Implications for design

Our observations of how graphic designers interact with layout suggest that visual composition tools should support the following:

- Creating links between visual properties:

Size, position, alignment and other spatial properties play an important role in creating the layout. They are especially important when the designer creates links between them.

Visual composition tools should facilitate creating these links, applying them to visual elements, and changing them as the structure evolves. This will help designers define fixed properties they want to keep unchanged. It will also help them define flexible properties they vary throughout the composition.

- Defining rules for dynamic properties:

Graphic designers define patterns manually. Alternatively, they use program functions, in environments like processing (Colubri and Fry 2012), to define how visual properties change in space and time. The programmatic approach is more efficient than the manual, repetitive application of these changes. However, writing visual rules as code omits the advantages of experimenting with these rules as they are created.

Visual composition tools should support creating rules that define how properties evolve in space, throughout the composition, or in time, across related compositions.

6.7 SUMMARY

Graphic designers establish and convey their organization concepts through visual properties. As they structure visual content, graphic designers identify fixed properties they keep unchanged. They also define flexible properties they change throughout the composition. Graphic designers also define rules for changing visual properties. They create these changes manually, as they establish temporal patterns, or programmatically, by defining rules and ranges for dynamic visual properties.

6.8 CONTRIBUTIONS

1. Method for extracting specific story elements from critical object interviews, and creating a representation of critical object stories: *StoryPortraits* with Nolwenn Maudet.
2. Found that graphic designers create links between visual properties. They create rules to define how these links change in time, across related compositions, and in space, across instances of the same composition.

3. Implications for the design of layout creation tools. Visual composition tools should support linking spatial properties. They should support creating rules to define how these properties change in space and time.

DESIGNING FOR MANIPULATING VISUAL PROPERTIES

This chapter presents interactive prototypes that demonstrate the feasibility of the design implications presented in the previous chapters. I argue that visual composition tools should: reify visual prototypes into first class objects. I present four color tools and two layout tools based on design recommendations extracted from three empirical studies with designers.

I then present two probe studies: one with the color tools, and another study with the layout tools. Users present the tools conceptually, by articulating what each tool does, and practically, by providing detailed examples of how these tools could be used in their current projects.

7.1 MANIPULATING VISUAL PROPERTIES: IMPLICATIONS FOR DESIGN

Studies with designers uncovered their innovations, workarounds, and novel strategies as they manipulate visual properties. We observed their practices when working on real projects (Chapter 5, Chapter 6), and when using a single application (Chapter 4).

As they manipulate color, users do not just select a color from the representation of a color model. They create personal representations of color, and base their selection on the information these representations communicate. Users interact with groups of color, in the context of their surrounding visual properties. They do so by changing the visual properties of color swatches, such as size, position and layering. They also manipulate new visual elements they create by combining color with other visual properties such as texture.

As they structure visual content, users do not only define fixed values for spatial properties such as size, position and alignment. Layout practices involve manipulating links between these properties throughout the composition. Users define fixed, and flexible visual properties to create unity and variety in their layout. They also create rules to define how flexible properties vary over time. They define these rules programmatically, or manually by redefining them for each instance in their composition.

As they perform visual composition tasks in an application, users manipulate properties directly in the composition. They prefer strong specific tools (Denning 2001), that provide direct access to these properties, over dialog boxes, that hide them behind graphical objects.

Based on these observations, we suggest directions for the design of tools to support manipulating visual properties for online visual composition.

- Reifying visual properties:

We use the term reification: “creating new objects that can be manipulated by the user, thus increasing the set of objects of interest” (Beaudouin-Lafon and Mackay 2000).

We argue that visual composition tools should turn visual properties from values attached to objects, hidden and accessed through dialog boxes, to first class objects (section 2.2.2).

Color manipulation practices involve manipulating the visual properties of color (section 5.7.4). Users perceive and manipulate colors as first class objects. They now use ad-hoc strategies to manipulate the visual properties of color. They create colored shapes for the purpose of manipulating their colors. They currently need to go back and forth between this representation they create on the document, and fixed, small color swatches provided by color tools, such as the color picker.

Visual composition practices involve manipulating links between spatial properties (section 6.6.2). This practice shows the benefit of reifying visual properties. Commands such as grouping, might apply to visual properties such as size. For example, users might link several graphical objects of the same size in the same group, and manipulate this link in their composition.

The reification of visual properties would also encourage thinking about these properties as independent objects. Thus, they would benefit from more consistent representations, and more consistent tools for interacting with these representations (section 4.6.2).

- Manipulating visual properties directly in the composition:

Users manipulate visual properties by changing their values for graphical objects in their document. We argue that the manipulation of visual properties should take place directly in the visual composition.

Color manipulation practices include changing the visual properties of color, such as size, position and layering, in the context of its surrounding properties. Thus, the values of other visual properties, including other colors, affect users’ color choices (section 5.7.4).

Visual composition practices include defining rules for visual properties that change in space or time. Users proceed using code functions or manually by specifying values for the visual property in each instance of the composition. When they use code, users miss the benefit of adjusting their rules instantly, depending on the result of the current manipulation (section 6.6.2).

In a composition application, users prefer tools that provide direct access to visual properties. They use composition strategies where they build the visual output by combining smaller visual elements, and manipulating visual properties at this scale. For example, to align 35 lines of text, they would align two lines, then four and so on (section 4.6.2).

7.2 MANIPULATING VISUAL PROPERTIES: INTERACTIVE PROTOTYPES

We present interactive prototypes that demonstrate the feasibility of using the recommendations above to design tools for manipulating visual properties.

7.2.1 *Color Prototypes*

We created a set of four prototypes, each designed to explore an activity that is not well supported by current tools. The prototypes demonstrate design implications related to color manipulation. Note that our goal was not to fully support all possible color manipulation activities, but rather to illustrate how color tools might inspire new directions for supporting color manipulation.

Two prototypes support creating personal color representations, and two prototypes support manipulating color in context. *Color Partner* visualizes the process of color manipulation. *Color Revealer* visualizes the progress of an activity through color. *Palette Explorer* supports manipulating the visual properties of color in context. *Color Composer* supports manipulating the combination of color and texture as a single visual element.

7.2.1.1 *Color Partner: Visualizing color manipulation*

Users often reuse previous material from earlier projects. They create personal color representations, to find colors for their current artifacts.

Color Partner lets users capture interim steps in their color exploration process. Users identify a starting point by specifying two col-

ors, after which *Color Partner* generates novel, related colors. In figure 69.a, the user creates a white and a black dot. In figure 69.b, the system generates a set of related dark colors.

Users can guide color generation by manipulating the distance to previously generated colors: moving the cursor closer to a dot produces more similar colors whereas moving the cursor away generates more diverse colors. In Figure 69.b, the user generates pale colors in the red range by moving from the red dot on the right, towards the white dot.

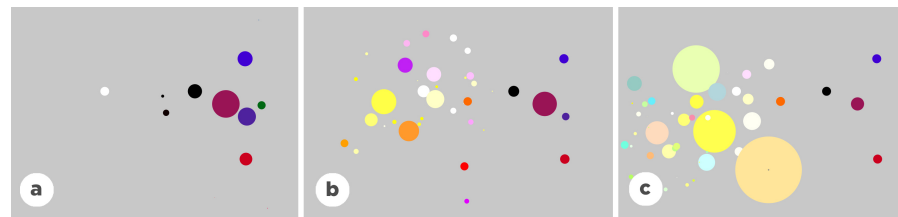


Figure 69: Color Partner: Users specify initial colors and move the cursor to control generation of new colors. They can save interesting intermediate colors by clicking on the dot.

Over time, unselected colors become smaller and disappear. However, the user can save colors by clicking on them; multiple clicks enlarge the size of the dot. In figure 69.c, the user indicates a strong preference for three yellow and pale orange colors. *Color Partner* allows users to return to intermediate color choices and use them to create new colors. *ColorPartner* offers an interesting partnership between users and the system, since they both collaborate to create informative personal color representations.

Color Partner supports creating personal, informative representations of the color manipulation process. Users can benefit from the cues the system provides (size of the dots, distance between the dots), to guide their color manipulation process.

7.2.1.2 *Color Revealer: Visualizing a process through color*

Color changes reveal intermediate steps or the overall state of an activity as it occurs over time.

Color Revealer does not treat color as an end in itself, but rather as a means to an end. Like *EditWear* (Hill et al. 1992), which uses color to support annotation, *Color Revealer* captures traces of the user's writing process, through changes in hue and intensity. Each character appears in a subtle, colored layer in the background.

In figure 70.a, the user writes with very little hesitation, so the layers behind the text appear in pale green. Additional layers appear as the user deletes or rewrites words. In figure 70.b, the user has repeatedly deleted and corrected a number of words in his introduction, which adds blue and purple layers on the text. *Color Revealer* allows users to control the mapping of colors to their writing activities. *Color Revealer* also provides an adjustable timeline that lets users scroll back through earlier stages in the writing process. In figure 70.c, the user needs additional time and has adjusted the time scale accordingly.

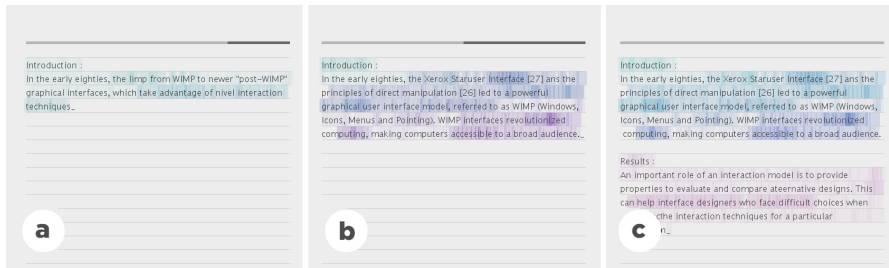


Figure 70: *Color Revealer*: Users' hesitations and corrections are revealed through changes in hue and intensity.

Color Revealer permits creating personal, informative color representations of how a process evolves. These representations are based on the progress of an activity over time. They contain information about the length of this activity and visually communicate its phases.

7.2.1.3 *Palette Explorer: Manipulating Color visual properties in context*

Users actively create sets of colors and manipulate them in the context of surrounding visual properties, including other colors in the palette. Users usually combine features from multiple tools to create ad-hoc strategies for these color manipulations.

Palette Explorer lets users create groups of multi-colored swatches of different sizes and shapes. In figure 71.a, the user creates a large green foreground swatch, with a pale blue background, and red and blue intermediate elements. The user can move, resize and adjust layers to explore the effects of different spatial relationships. In figure 71.b, the user changes the background to red and the remaining elements are smaller and repositioned in layers next to each other.

Users can modify a color in the context of the surrounding colors by moving the cursor along three axes: X for hue, Y for saturation and mouse wheel for brightness. Users can also select sets of swatches or the whole palette and modify them at the same time, retaining the harmony and other characteristics of the original palette. In Fig-

ure 71 c, the user changes the main foreground blue to orange. The remaining palette colors follow the same change. *Palette Explorer* sup-

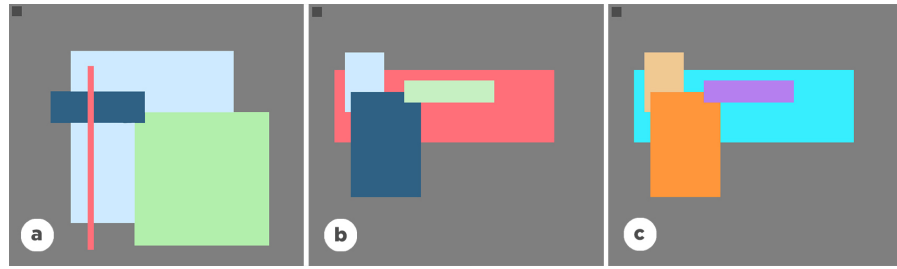


Figure 71: *Palette Explorer*: Users create interactive color palettes with dynamic swatches. They can manipulate the visual properties of color swatches in context.

ports manipulating visual properties of color, such as size, position and layering. Users can change these properties in the context of the surrounding colors. Through manipulating these properties, they can create personal color representations using specific values for color visual properties.

7.2.1.4 *Color Compositor: Manipulating the combination of colors and textures*

Some color pickers offer users a limited set of patterns or textures (Pang 2010). These color pickers do not permit the manipulation of the result of combining a color and a texture, as one visual element.

Color Compositor lets users combine colors and textures, and manipulate their own composites. In Figure 72 a, the user combines orange with an image containing slices of a citrus fruit. She creates a new visual element, through combining a color and a texture. In Figure 72 b, the user turns the orange into a lemon. Users can also decompose the resulting textured image into its component parts. In Figure 72 c, the user extracts color and texture for later reuse.

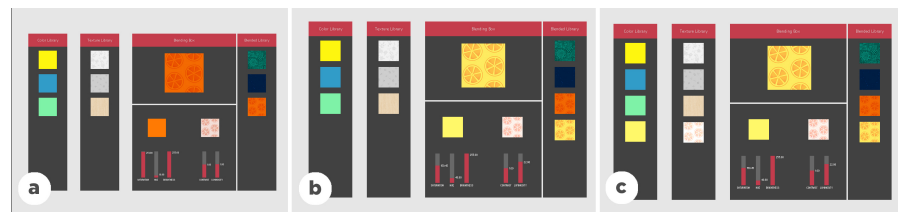


Figure 72: *Color Compositor*: Users combine colors and textures to manipulate composites.

Color Compositor combines colors and textures. It supports manipulating the visual properties of this combination. Users can create this combination visually on the colors and textures of interest. *Color Compositor* also supports extracting color and texture from a colored-texture, and getting its component elements.

7.2.2 *Study: Using Color Prototypes*

We wanted to find out how users interpret each tool in general, and in the context of their current work. We run a study where participants try each prototype, and answer guiding questions about their interpretation of the tools purpose, and of its utility in their current projects.

7.2.2.1 *Participants*

We interviewed eight participants (5 men, 3 women, aged 23- 40). Professions included: product designer, illustrator, painter, service designer, researcher in data visualization, information theorist, virtual reality engineer and programmer.

7.2.2.2 *Procedure*

Each session lasted approximately one hour in the participant's studio or office. We presented each tool, in turn, and gave the participant five minutes to experiment and perform short tasks with each tool, as follows: Palette Explorer: Create a book's cover page. Color Compositor: Design a textured color for the background of the book cover. Color Partner: Create your favorite red and favorite blue. Color Revealer: Write a summary of a recent project. After each task, we asked participants to think of recent color projects in which they manipulated color and show us how the tool might, or might not, be useful for those tasks. We counter-balanced the order of tools across participants.

7.2.2.3 *Data Collection*

We collected audio recordings of each session and screen captures of their interactions with each color tool. We also took notes based on participants' answers to a common set of interview questions.

7.2.2.4 *Results*

Participants suggested different ways of using the color prototypes both conceptually, by explaining what the tool does, and practically, to solve problems for their current projects. Interestingly, all participants wanted to interact with the color tools immediately, to manipu-

late swatches, dots or textures to obtain a desired effect.

COLOR PARTNER Participants viewed *Color Partner* as a space for creating personal colors. P1 (Painter) appreciated how suggesting colors helped him make color-related decisions. However, he *“would not share the result of generation. The colors I get are more personal.”* *Color Partner* also helped participants create interesting clusters of colors and preserve those they would later use to create a new color exploration chain. P8 (Information theorist) felt that the tool would help her get *“little by little to the color I want.”*

COLOR REVEALER *Color Revealer* encouraged participants to reflect on their typing and writing practices. P8 (Information theorist) compared *Color Revealer* to an eraser: *“When I erase an area several times, I end up seeing the trace on paper.”* She also noticed that she always deleted the whole word when she makes a typo, which she had not realized before. P5 (Product Designer) explained that *Color Revealer* *“projects your thinking as you write. It helps you feel the intensity of your writing and gives it meaning through making this process visible.”*

All but one participant described how they would use *Color Revealer* in their current projects. P1 (painter) wanted to use it when collaborating on a course he is teaching. Since his colleague is a fast typist, he wanted her to use the tool to take notes during class. He would like to read her notes later and use the change of color to help him understand when she hesitated, just as in a face-to-face conversation.

PALETTE EXPLORER Participants (7/8) viewed *Palette Explorer* as a space for sketching color relationships relevant to those in their daily work. P5 (Product designer) described *Palette Explorer* as *“a tool for projection and sketching that allows rapid visualization of scale relationships among colors.”* She wanted to create a color chart to establish a digital identity for a client and was particularly interested in using it to see how the colors look together in different contexts. For example, she wanted to establish a set of colors and then experiment with different versions of a client’s logo.

COLOR COMPOSITOR Half of the participants reported developing ad-hoc strategies to combine colors and textures, but reported that *Color Compositor* greatly simplified the task. P3 (Service designer) was happy to avoid using Photoshop layers when creating and experimenting with a textured color. P5 (Product designer) wanted to use *Color Compositor*’s texture library to group similar components for future use. Although *Color Compositor* focused on texture, participants had a number of suggestions for combining color with other elements, such as images. They also offered innovative ideas for how

to extend the tool. For example, three participants suggested reproducing these patterns on very large surfaces, to establish color and texture relationships when creating wallpaper.

Participants reported on potential uses of the four prototypes. Their stories amplify the importance of creating personal, informative color representations. These stories involve manipulating color visual properties and combining color with other visual properties. Participants also explained how the context of their specific projects affects their decisions.

7.2.3 *Manipulating visual properties to structure visual content*

We present two prototypes, each designed to explore a visual composition practice current tools do not fully support. The prototypes represent different design implications related to visual composition. Note that our goal was not to fully support all possible composition activities, but rather to illustrate how tools might support particular practices in visual composition.

7.2.3.1 *Linkify: Manipulating links between spatial properties*

Users create links between spatial properties such as size, position and layering (section 6.6.2).

Linkify supports creating links between content properties. The user selects two content elements and the system captures the visual ratios between two of their properties. For the purpose of the prototype, we implemented a small set of properties: the width of the element and its horizontal and vertical position, and the font size and number of characters of text elements. Once the link is created, it is automatically applied when the content elements change, like a spreadsheet recomputing formulas when a cell changes.

We foresee two main benefits to this approach. Since the layout relies on content properties, it will automatically take into account new content and visually evolve accordingly. Also, the designer can anticipate the outcome of the layout by directly testing it with new content, but may also let herself be surprised by the generated layouts, encouraging serendipitous exploration.

We illustrate this prototype with a simple scenario. Alice is a graphic designer. She wants to design the layout for a blog. The author gave her the three first articles. Alice decides that the body text will remain a stable element for the reader, but wants all the other elements to generate a new composition for each new blog post. First, she creates

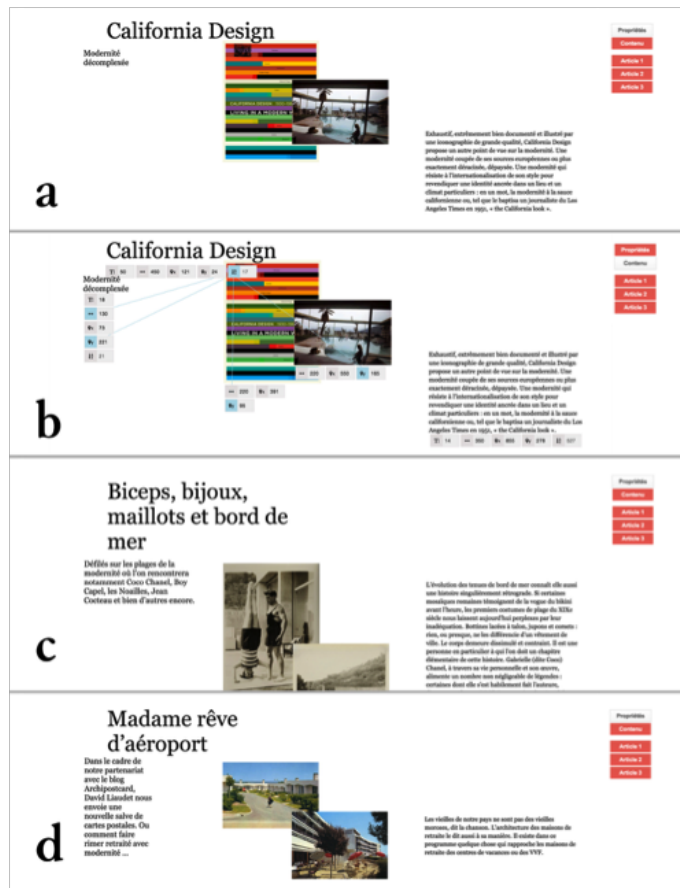


Figure 73: *Color Compositor*: Users combine colors and textures to manipulate composites.

an interesting composition with her content (figure 73.a). She then visually and directly links the title character length parameter to the vertical positions of the images and of the subtitle (figure 73.b): The more characters in the title, the lower the other elements are going to be. Linkify automatically calculates the ratio based on the linked parameters. She then directly checks that the layout produces interesting results with both longer (figure 73.c) and shorter titles (figure 73.d).

Linkify supports linking visual properties directly on the document. It supports visually defining rules for how these links evolve as the content changes.

7.2.3.2 *Contextify: Dynamic visual properties*

Users define rules for visual properties that change in time, across compositions, or in space, in different instances of the same composition. They currently define rules using code. The alternative approach is to specify values of corresponding properties in each instance of their composition. The programming approach is more efficient. But

it omits the possibility of creating these rules based on visual experimentation. The manual approach allows slow-building the rules but is too repetitive.

Contextify supports visually defining rules for dynamic visual properties. The system generates the layout by combining sub-layouts created with each rule. For the purpose of the prototype, we implemented two sets of rules. The first rule is contextual: the designer specifies values for properties for day and night. The second rule depends on the reader: the designer specifies values of properties when the reader accesses the web page in preview or detailed reading mode.

We illustrate this prototype with a simple scenario. Bob wants to design the layout for an article to appear in an online magazine. For the preview condition, Bob wants to show an overview of the article. He emphasizes the title size to attract attention and includes a selection of text and images (figure 74.a). He saves this first sub-layout. To guide the reader through the content in the detailed reading condition, he creates a diagonal flow (figure 74.b) and saves the second sub-layout. Bob decides to change colors for the night and day conditions. He selects two colors (figure 74.c.d). The reader can now access different versions for the same article, depending on the rules Bob created.

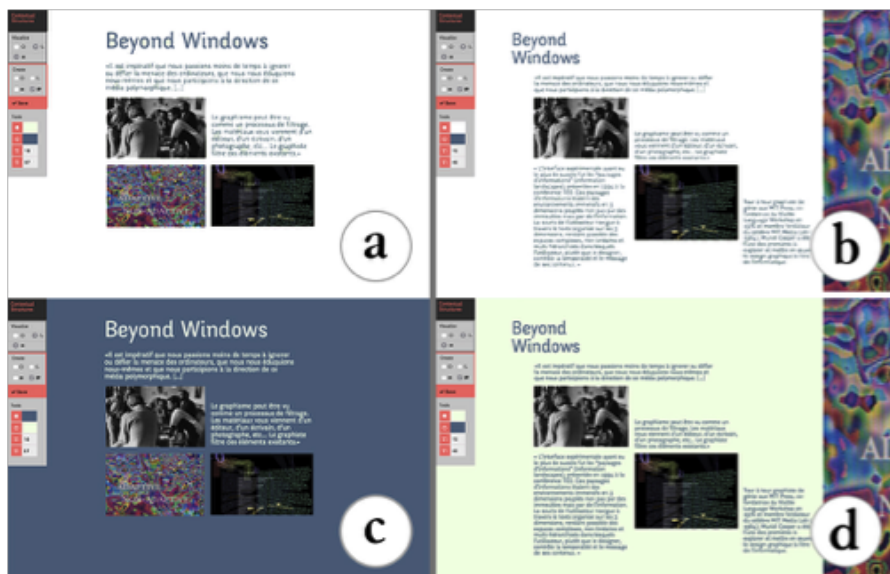


Figure 74: *Color Compositor*: Users combine colors and textures to manipulate composites.

Contextify supports creating rules that define how properties change throughout a visual composition. It also supports creating these rules directly on the content elements.

7.2.4 Study: Using Layout prototypes

We wanted to find out how users interpret each tool in general, and in the context of their current work. We run a study where participants try each prototype, and answer questions about their interpretation of the tool's purpose, and of its utility in their current projects.

7.2.4.1 Participants

We interviewed 12 graphic designers (6 men, 6 women), age 23-57, with 4-27 years of experience (mean=11), who work in various environments (freelance, studio, agency) and create layouts for digital media (n=1) or both digital and print media (n=11).

7.2.4.2 Procedure

Each session lasted approximately one hour and a half. We gave a scripted presentation of the functionalities of each tool and gave the participant 10 to 15 minutes to perform a short task based on the following scenarios (figure 75). They could then continue to experiment with each tool.

In *Linkify*, the task was to use three sample articles and create a layout for a graphic design online blog based on the title's length. In *Contextify*, the task was to create a layout for an article in an online design magazine that adapts for daytime and night time, and to the reader's choice of getting a preview or the full article.



Figure 75: *Color Compositor*: Users combine colors and textures to manipulate composites.

We used a think-aloud protocol and we counterbalanced the order of tools across participants. After each task, we asked participants to

describe to one of their colleagues what they think was the purpose of the tool and how to use it. We also asked them if they had a recent layout project in which they thought the tool could have been useful and to describe in detail how the tool could have been used for this project. We also asked about possible improvements to the tool in the context of that project.

7.2.4.3 Data Collection

We collected audio recordings of each session and screen captures of their interactions with each layout tool. We also took notes based on participants' answers to our questions.

7.2.4.4 Results

Participants suggested different ways of using the prototypes both conceptually, by explaining what the tool does, and practically, to solve problems for their current projects.

LINKIFY

Eleven (11/12) participants reported stories that involve creating links between the properties of visual elements. Eight participants suggested improvements to the current layouts involved in the task, to use them in their current projects. P11 sees the tool as a way to save time when she designs similar presentation slides. She would use the tool to *"avoid manually adjusting each page"* since the tool adjusts the layout to the content properties. P8 would use the tool to rapidly explore layouts for school books: *"their structures are very rigid, but they have to adapt to diverse content elements."*

More than half of the participants (7/12) wanted to use the tool to create layouts they could not create with traditional composition tools. For example, P1 would like to use the tool *"to create a completely different layout without having to redesign everything from scratch. With Linkify, all pages evolve when I modify the last page."*

P5 would use *Linkify* to remember the fixed properties of all his layouts. The system could directly reuse these properties at the beginning of a new project. P7 would use *Linkify* for teaching graphic design. He would add links to new properties such as opacity. The students will establish links between content properties, and visualize how the layout changes as the content changes. Other participants also wanted to add additional properties such as white space between elements (P3, P5, P11).

Participants wanted to access other properties such as margins and visual points of reference (P5). They also wanted new ways of creating links between properties. For example, they wanted to manipulate ratios by inverting them or to define absolute and relative ratios (P7). They also wanted to define the bounds and extreme values (P4, P1) of some properties.

CONTEXTIFY

Ten participants reported stories that involve visually defining rules for dynamic properties using *Contextify*. Participants also suggested improvements to the current prototype, and ways to use it in their current projects.

Nine participants wanted to use the tool to create layouts they could not create with their traditional tools. For example, P8 wanted to use the tool to design school content on tablets. *Contextify* provides multiple layouts that could adapt to the learning methods of different students: *“Some students need more images, and others need more words. Other students need to see all the content at a glance. It would also be very interesting to add specific types of content only at home, such as sound for an English workbook”*. Similarly, P4 explained that *“the tool could help create and link a global view of her layout and a very detailed one”*.

Participants reported on potential uses of the two prototypes. Their stories show the importance of creating links between properties. They also provide examples of using the tools to visually define rules for dynamic properties.

7.3 SUMMARY

In this chapter, I argue that visual composition tools should: reify visual prototypes into first class objects.

I present four color tools and two layout tools based on design recommendations extracted from three empirical studies with designers.

I then present two probe studies: one with the color tools, and another study with the layout tools. These studies follow the same method. Users present the tools conceptually, by articulating what each tool does, and practically, by providing detailed examples of how these tools could be used in their current projects.

7.4 CONTRIBUTIONS

1. Design and implementation of interactive color prototypes with Nolwenn Maudet.

2. Found that two layout prototypes, designed by Nolwenn Maudet, demonstrate the possibility of creating rules for links between visual properties.
3. Found that designers identify with the interactive prototypes. They explain how they can use these prototypes in their current visual compositions.

CONCLUSION

To conclude this work, this chapter summarizes the contributions of the thesis and presents future directions for research.

8.1 THESIS SUMMARY

Chapter 2 introduces the main concepts used in this thesis. First, it presents a framework for refiying visual properties to support composition tasks. Then, it presents factors affecting how designers create their visual compositions. Perception, design tools and guidelines affect how designers create their compositions. Then, it describes visual composition as a reflective practice, where designers know more than they say, and respond to the situated circumstances of creating their artifacts. Chapter 3 analyzes the main approaches composition tools propose for interacting with properties. Several composition tools follow an approach where properties are inferred from user's actions on the object. This approach provides direct access to the object, but no representation for its properties. Other tools suggest enhancements to the traditional property sheets, adding new ways for changing object properties through dialog boxes. A third approach consists of adding a layer, on the object, where the user can manipulate its properties. The fourth approach, and closest to responding to our framework, creates an independent representation, with specific interactions, for object properties in the document.

Chapter 4 focuses on graphic designers' visual composition practices within a specific application. I present recurrent strategies across designers and the types of tools they use. Based on this comparison, I suggest new directions for the design of online visual composition tools. As they organize content, designers change object properties on the document, instead of using dialog boxes. Chapter 5 and chapter 6 focus on designers' color and layout manipulation practices, as they create their artifacts. We extract and analyze stories from interviews with designers about their practices in real projects. Based on these observations, these chapters summarize main color and layout practices, and the resulting implications for the design of color and layout tools. Designers create personal representations of color. They use these representations to find color schemes for their artifacts. Designers create links between object properties such as color, size and position. They also create rules to define how these links evolve in

space, in the same document, or in time, across related documents.

Chapter 7 presents tools that illustrate the reification of visual properties into first class interactive objects. We introduce two groups of prototypes. Four color tools demonstrate new possibilities for color manipulation. These tools support creating personal representations, changing the properties of color swatches, and combining color with other properties, such as texture. Two layout tools support creating links between spatial properties such as size and position, and defining rules for dynamic properties. Two studies present how designers used and described the tools with respect to their color and layout practices.

8.2 CONTRIBUTIONS

This thesis provides contributions at three levels, with empirical findings, illustrative tools and theoretical perspectives. I summarize them here, following the dissertation plan.

8.2.1 *Theoretical perspectives: Framework for reifying visual properties for composition*

Designers base their visual composition practices on their knowledge of visual organization principles. They also base their composition practices on how they perceive the objects and on the design tools they use to create their artifacts. These factors guide, but do not dictate how the visual composition looks like at the end. Visual concepts depend on the designer's reflective conversation with the design materials. For this reason, understanding designers' composition practices requires observing them as they create their artifacts.

Based on the observation and analysis of designers' composition practices, I propose a framework that describes how composition tools can support the reification of visual properties, by treating them as first-class interactive objects. I argue that the reification of visual properties into first-class interactive objects requires supporting:

- Personal, instead of generic representations of visual properties.
- Interactions with visual properties in the composition, not in separate documents or windows.
- Links among visual properties of the same type, and among visual properties of different types.

- Rules for evolving visual properties in time, across several compositions, and in space, across components of the same composition.

8.2.2 *Empirical Findings: Designers' composition practices*

Designers use visual properties, such as color, when they create their artifacts. They create personal color representations instead of using the generic representations current color tools provide. Designers use personal, informative representations as a starting point for color selection. They create these representations based on collected samples from previous artifacts, collected images, or standard color spaces. Designers manipulate the visual properties of color. They change the size, position and layering of individual color swatches. Designers also combine color and other properties such as texture, and use this combination as a single visual element.

Graphic designers use spatial properties such as size, position and layering, to organize visual content. They create links between spatial properties and manipulate these links throughout their compositions. As they create their layouts, graphic designers define fixed and flexible properties. They create rules to define how flexible properties change in space, across instances of the same composition, or in time, across related compositions.

Graphic designers complete visual composition tasks through direct manipulation of visual properties in the document. They use tools that act directly on these properties. They also find novel ways to use properties as tools for composition tasks.

8.2.3 *Technologies: Designing for the manipulation of visual properties for composition tasks*

We propose color and layout tools that illustrate how designers could reify visual properties into first-class interactive objects. *Color Partner* supports creating personal, informative representations of the color manipulation process. Users can benefit from the cues the system provides (size of the dots, distance between the dots), to guide their color manipulation process. *Color Revealer* permits creating personal, informative color representations of how a process evolves. These representations are based on the progress of writing over time. They hold information about the length of this activity and visually communicate its phases. *Palette Explorer* supports manipulating visual properties of color, such as size, position and layering. Users can change these properties in the context of the surrounding colors. Through manipulating these properties, they can create personal color repre-

sentations using specific values for color visual properties. *Color Compositor* combines colors and textures. It supports manipulating the visual properties of this combination. Users can create this combination visually on the colors and textures of interest. *Color Compositor* also supports extracting color and texture from a colored-texture, and getting its component elements.

Linkify supports linking visual properties directly on the document. It supports defining how these links evolve as the content changes. *Contextify* supports creating rules that define how visual properties change throughout a visual composition. It also supports creating these rules directly on the content elements.

8.3 LIMITATIONS AND PERSPECTIVES

We proposed a framework based on designers' practices in visual composition. This framework presents how visual composition tools can transform properties into first-class objects. The current dimensions include accessing properties in the document. They also include creating links between these properties, defining rules for changing properties, and creating personal representations of visual properties. We based these dimensions on observing graphic designers' digital and physical composition practices. Further studies of designers' practices could contribute with new dimensions to this framework. They can reveal new ways of representing and using properties for visual composition tasks.

Our findings are based on observing the practices of a small number of experts, with specific skills. Our goal is to find emerging phenomena from recurring, but also surprising observations. Our findings suggest new opportunities for testing more specific hypotheses using more quantitative experimental approaches. For example, controlled studies like (Mackay 2002a) can investigate which interaction techniques are better suited for the practices we identify.

I focused on color manipulation and layout creation practices. Future work can investigate practices in manipulating other visual properties. For example, observing the practices of typographers as they create new typefaces, might provide interesting insights about how they deal with the duality of meaning and form when interacting with type.

In this thesis, I focused on the manipulation of visual properties in desktop environments. One potential direction would be investigating the benefits and challenges experts face as they manipulate visual properties in other environments, such as touch interfaces. In these interfaces, property sheets are not practical (Xia et al. 2016).

Thus, composition tools rely on direct manipulation, using a limited vocabulary of gestures. Interesting questions relate to extending the current gesture vocabulary to support visual composition using gestural interaction.

We propose tools that illustrate separate dimensions in the framework of reifying visual properties for composition tasks. Further work on the tools could focus on investigating how they communicate and interact. This would permit creating more comprehensive visual composition tools that better support the whole composition process, instead of separate composition practices.

Our findings open possibilities to create new tools based on the suggested design implications. We want to inspire composition tool designers to create tools that respond to the composition practices we identify.

CRITICAL OBJECT INTERVIEWS: COLLECTED DATA

- Video recordings:
We use video recordings to capture demonstrations and participants' interactions with tools and artifacts in their work environments.
- Photographs:
We use photos to capture participants' work settings, tools, sketches and artifacts (Fig. 76).



Figure 76: Photo of a participant's tools and artifacts in her workspace.

- Written notes:
We use the interviewers' hand-written and typed notes as a starting point to extract critical object stories. They provide summary points, sketches, and surprising elements from the interviews.
- Audio conversations:
We use the audio recording to extract participant quotes and articulations of specific aspects of their manipulation practices. For example, a participant described her process using a color picker saying :“I purify the color by removing the black using the CMYK sliders”.

Critical object stories gather data from these heterogeneous streams. They summarize relevant elements from the participant work process when creating an artifact. Specifically, a critical object story typically includes the following components:

- **Story Title:** We give the critical object story a title that summarizes and communicates the main concept in the story, a surprising aspect, a breakdown, or a word the participant used.
- **Story Code:** We give each critical object story a code. Typically, the code includes the participant number and the story number. This helps us refer to the story in the following phases of the analysis.
- **Story Steps:** We extract from the story an exhaustive set of steps that communicate the story's content to a reader who did not take part in the interview.
- **Quotes:** We include the participant's quotes along with contextual information to situate them in the story.
- **Tools:** When possible, we identify tools the participant used in the story. These tools could be physical, like a pen or a paintbrush. They could be features within an application, like the Adobe Photoshop pen tool. They could also be applications, like Adobe Color ¹

¹ <https://color.adobe.com/>

COLOR PORTRAITS: INTERVIEW QUESTIONS

B.1 INTERVIEW OBJECTIVES

We want to collect stories about specific manipulations of color in the context of a project or of a process where the participant was involved either alone or within a group. We are mainly interested in learning about how people with different backgrounds and goals want to manipulate color and how they actually end up doing that in their daily work. We are looking for details related to specific manipulations of color. We ask participants to show us previous projects, preferably recent projects and to tell us how they manipulated color for this specific project. We make sure the interviewees have access to their previous projects and that they can show us elements from these projects if they want to.

B.2 INTERVIEWEE RECRUITMENT

We chose people who manipulate color excessively in their work. We want people who have a process of activity that includes extensive manipulation of color.

B.3 INTERVIEW PROCEEDING

Before starting the interview, we tell the participant the following introduction: "We are working on a project on how people manipulate color and we would like to collect stories about how people who manipulate color extensively interact with color in their respective fields". We turn on the audio recording and we let the person show us previous projects and walk us through how she proceeded for selecting colors for these projects. We probe for more details and ask more questions to get specific details about how the user proceeded in the particular case of this project. We take pictures of the projects (artifacts) or short videos of the participant interacting with the object. After the interview, we stop the audio recording, we thank the participant and we provide our contact information in case they want to see the results of our work.

B.4 DEBRIEFING

After each interview, we go through the audio and written notes and we extract stories from the interview. We make sure we include the

most surprising elements in each story and we include as many details as possible. The debriefing is also the time to point out the most surprising points and to compare them in order to identify the new and unusual color manipulation practices in each story.

GROUNDED THEORY CATEGORIES

C.1 TYPOLOGIES OF STRUCTURES:

Structures types are non exclusive, they can intersect

C.1.1 *Semantic structures:*

Structures defined using semantic properties of the content
Intrinsic: Structures defined using inherent semantic properties of content elements
Extrinsic: Structures defined using semantic relationships among content elements

C.1.2 *Spatial structures:*

Structures defined by spatial properties of the graphical elements.
Fill: Structure defined by elements' visual masses (visual weight, negative space)
Border: Structures based on elements borders or other lines (ex : grid, guides)
Size: Structures defined by the size of the elements.
Position: Structures defined by relative positions of elements.
Layering: Structures defined by superpositions of elements. (ex : layers)

C.1.3 *Pattern structures:*

Structures defined by the repetition of graphical elements or pre-existing structure in time, in space or in both. (ex: navigation principle, rhythm)

C.1.4 *Conceptual structures:*

Structures defined using principles that can be either specific or general.
Specific: Structures defined using clearly identified and defined rules or constraints. These rules could be executed by the system or by the user.
General: Structure defined using a non specific concept. (ex: style, ambiance, idea).

C.1.5 *Example structures:*

Structures defined using examples.
Usual: Structures defined by normal or recurring examples from the final document.
Exception: Structures defined using extreme or exceptional examples from the final

document. Placeholder: Structures defined using a user-generated or system-generated artificial example (ex: filler text (lorem ipsum)).

C.2 INTERACTION WITH STRUCTURES:

How do users create, manipulate or use different types of structures ?

C.2.1 *Reuse/Application:*

Ways of reusing/applying a structure Iterate/Develop : Creating a variant structure based on an existing one by varying only part of the components of the source structure. Propagate : Propagating structure property Combine : Combining several structures

C.2.2 *User-System Partnership:*

Ways of collaborating in the structure creation, manipulation or application Manual: The user must apply or use the structure fully manually Collaborative: The system either participates in the structure creation, manipulation or application

C.2.3 *Adaptability:*

Ways of adapting a structure Flexible: The structure has the affordance to be modified. Hackable: The structure doesn't have the affordance to be modified but it can be overridden or its properties can still be modified. Unbreakable: The structure cannot be overridden, neither can it be modified.

C.2.4 *Informing Content:*

The structure is used to reveal new information about the content

C.2.5 *Filtering*

The structure is used to filter the content

BIBLIOGRAPHY

- Adobe. 2000. *The RGB (CMY) Color Model - Color Models - Technical Guides*. Visited on 08/23/2016. (Cit. on pp. 28–30).
- Adolph, Karen E., and Kari S. Kretch. 2015. “Gibson’s Theory of Perceptual Learning.” In *International Encyclopedia of the Social & Behavioral Sciences (Second Edition)*, ed. by James D. Wright, 127–134. Oxford: Elsevier. Visited on 05/19/2015. (Cit. on pp. 11, 17, 18).
- Albers, Joseph. 1963. *Interaction of Color: 50th Anniversary Edition*. Anv edition. New Haven, CT: Yale University Press. (Cit. on pp. 22, 27, 31, 32).
- Apple Computer, Inc, ed. 1992. *Macintosh human interface guidelines*. Apple technical library. Reading, Mass: Addison-Wesley Pub. Co. (Cit. on p. 48).
- Arnheim, Rudolf. 1974. *Art and Visual Perception: A Psychology of the Creative Eye*. Google-Books-ID: Z7LQRiSM1I4C. University of California Press. (Cit. on p. 37).
- Bauersfeld, Penny F., and Jodi L. Slater. 1991. “User-oriented color interface design: direct manipulation of color in context.” In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 417–418. ACM. Visited on 09/13/2014. (Cit. on p. 85).
- Baumann, Narciso Silvestrini, and Stromer Verlag. 1994. *Isaac Newton. color system: color systems in art and science*. Visited on 08/18/2016. (Cit. on p. 26).
- Baxter, Isobel, and Keith Oatley. 1991. “Measuring the learnability of spreadsheets in inexperienced users and those with previous spreadsheet experience.” *Behaviour & Information Technology* 10, no. 6 (): 475–490. Visited on 09/13/2015. doi:10.1080/01449299108924305. (Cit. on p. 66).
- Beaudouin-Lafon, Michel. 2000. “Instrumental interaction: an interaction model for designing post-WIMP user interfaces.” In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 446–453. ACM. Visited on 10/02/2014. (Cit. on p. 12).
- . 2004. “Designing interaction, not interfaces.” In *Proceedings of the working conference on Advanced visual interfaces*, 15–22. ACM. Visited on 10/02/2014. (Cit. on p. 63).
- Beaudouin-Lafon, Michel, and Wendy E. Mackay. 2000. “Reification, polymorphism and reuse: three principles for designing visual interfaces.” In *Proceedings of the working conference on Advanced visual interfaces*, 102–109. Visited on 01/18/2014. (Cit. on pp. 12, 13, 112).

- Berryman, Gregg. 1990. *Crisp: Notes on Graphic Design and Visual Communication*. 3 edition. Los Altos, Calif: Crisp Learning. (Cit. on p. 65).
- Bier, Eric A., Maureen C. Stone, Ken Pier, William Buxton, and Tony D. DeRose. 1993. "Toolglass and magic lenses: the see-through interface." In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, 73–80. ACM. Visited on 09/30/2014. (Cit. on pp. 52, 61).
- Blackwell, Alan F. 2006. "The reification of metaphor as a design tool." *ACM Transactions on Computer-Human Interaction (TOCHI)* 13 (4): 490–530. Visited on 05/19/2015. (Cit. on p. 48).
- Bringhurst, Robert. 2002. *The Elements of Typographic Style*. 2nd edition. Point Roberts, WA: Hartley & Marks Publishers. (Cit. on pp. 33, 34).
- Brown, Polly S., and John D. Gould. 1987. "An experimental study of people creating spreadsheets." *ACM Transactions on Information Systems* 5, no. 3 (): 258–272. Visited on 09/13/2015. doi:10.1145/27641.28058. (Cit. on p. 66).
- Buxton, William. 1986. "Chunking and phrasing and the design of human-computer dialogues." In *IFIP Congress*, 475–480. Visited on 06/08/2016. (Cit. on p. 65).
- Carter, Rob, Ben Day, Philip B. Meggs, Sandra Maxa, and Mark Sanders. 2015. *Typographic design: form and communication*. 6th edition. Hoboken, New Jersey: John Wiley & Sons, Inc. (Cit. on pp. 34, 35).
- Chevalier, Fanny, Pierre Dragicevic, and Christophe Hurter. 2012. "Histomages: fully synchronized views for image editing." In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, 281–286. Visited on 01/18/2014. (Cit. on p. 59).
- Colby, Grace. 1992. "Maintaining legibility, structure, and style of information layout in dynamic display environments." In *Posters and short talks of the 1992 SIGCHI conference on Human factors in computing systems*, 73–74. ACM. Visited on 06/29/2015. (Cit. on p. 50).
- Colubri, Andres, and Ben Fry. 2012. "Introducing Processing 2.0," 1. ACM Press. Visited on 08/05/2015. doi:10.1145/2343045.2343061. (Cit. on p. 108).
- Corbin, Juliet, and Anselm Strauss. 2007. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. 3rd edition. Los Angeles, Calif: SAGE Publications, Inc. (Cit. on pp. 4, 6, 88).
- Cullen, Kristin. 2007. *Layout Workbook: A Real-world Guide to Building Pages in Graphic Design*. Google-Books-ID: d2M_I7EXuoUC. Rockport. (Cit. on pp. 35, 36).

- . 2012. *Design Elements, Typography Fundamentals: A Graphic Style Manual for Understanding How Typography Affects Design*. Beverly, Massachusetts: Rockport Publishers. (Cit. on p. 33).
- Denning, Peter J., ed. 2001. *The Invisible Future: The Seamless Integration of Technology Into Everyday Life*. 1st edition. p145–179. New York: McGraw-Hill Companies. (Cit. on pp. 74, 111).
- Douglas, Sarah A., and Arthur E. Kirkpatrick. 1999. “Model and representation: the effect of visual feedback on human performance in a color picker interface.” *ACM Transactions on Graphics (TOG)* 18 (2): 96–127. Visited on 07/01/2014. (Cit. on p. 85).
- Dragicevic, Pierre, Stéphane Chatty, David Thevenin, and Jean-Luc Vinot. 2005. “Artistic resizing: a technique for rich scale-sensitive vector graphics.” In *Proceedings of the 18th annual ACM symposium on User interface software and technology*, 201–210. ACM. Visited on 03/05/2015. (Cit. on p. 54).
- Edge, Darren, Sumit Gulwani, Natasa Milic-Frayling, Mohammad Raza, Reza Adhitya Saputra, Chao Wang, and Koji Yatani. 2015. “Mixed-Initiative Approaches to Global Editing in Slideware,” 3503–3512. ACM Press. Visited on 05/05/2015. doi:[10.1145/2702123.2702551](https://doi.org/10.1145/2702123.2702551). (Cit. on p. 50).
- Edward R. Landa and Mark D. Fairchild. 2005. “Changing Color from the Eye of the Beholder.” *American Scientist* 93:436–443. Visited on 08/23/2016. (Cit. on pp. 28, 29).
- Evans, Poppy, and Mark A. Thomas. 2012. *Exploring the Elements of Design*. 3 edition. Cengage Learning. (Cit. on pp. 33, 36, 37).
- Faul, Franz, and Vebjørn Ekroll. 2002. “Psychophysical model of chromatic perceptual transparency based on subtractive color mixture.” *JOSA A* 19 (6): 1084–1095. Visited on 09/13/2014. (Cit. on p. 85).
- Felice, Marianela Ciolfi, Nolwenn Maudet, Wendy E. Mackay, and Michel Beaudouin-Lafon. 2016. “Beyond Snapping: Persistent Tweakable Alignment and Distribution with StickyLines.” In *Proceedings of the 29th annual ACM symposium on User interface software and technology*, to appear. UIST ’16. Tokyo, Japan: ACM, (cit. on p. 60).
- Flanagan, John C. 1954. “The critical incident technique.” *Psychological Bulletin* 51 (4): 327–358. doi:[10.1037/h0061470](https://doi.org/10.1037/h0061470). (Cit. on p. 5).
- Frisch, Mathias, Ricardo Langner, and Raimund Dachsel. 2011. “Neat: A Set of Flexible Tools and Gestures for Layout Tasks on Interactive Displays.” In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, 1–10. ITS ’11. New York, NY, USA: ACM. Visited on 09/15/2015. doi:[10.1145/2076354.2076356](https://doi.org/10.1145/2076354.2076356). (Cit. on p. 60).

- Frisch, Mathias, Sebastian Kleinau, Ricardo Langner, and Raimund Dachsel. 2011. "Grids & guides: multi-touch layout and alignment tools." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1615–1618. ACM. Visited on 06/29/2015. (Cit. on p. 60).
- Galletta, Dennis F., Kathleen S. Hartzel, Susan Johnson, Jimmie Joseph, and Sandeep Rustagi. 1996. "An experimental study of spreadsheet presentation and error detection." In *System Sciences, 1996., Proceedings of the Twenty-Ninth Hawaii International Conference on*, 2:336–345. IEEE. Visited on 02/01/2015. (Cit. on p. 66).
- Garcia, Jérémie, Theophanis Tsandilas, Carlos Agon, and Wendy E. Mackay. 2014. "Structured observation with polyphony: a multifaceted tool for studying music composition," 199–208. ACM Press. Visited on 09/15/2016. doi:10.1145/2598510.2598512. (Cit. on pp. 4, 6).
- Gibson, James Jerome. 1979. *The ecological approach to visual perception*. Houghton Mifflin. (Cit. on pp. 11, 17).
- Goethe, Johann Wolfgang von. 1970. *Theory of Colours*. New edition. London: MIT Press. (Cit. on pp. 23–26).
- Greeno, James G. 1994. "Gibson's affordances." Visited on 01/18/2014. (Cit. on p. 17).
- Halasz, Frank, and Thomas P. Moran. 1982. "Analogy Considered Harmful." In *Proceedings of the 1982 Conference on Human Factors in Computing Systems*, 383–386. CHI '82. New York, NY, USA: ACM. Visited on 08/04/2016. doi:10.1145/800049.801816. (Cit. on p. 48).
- Harrower, Mark, and Cynthia A. Brewer. 2003. "ColorBrewer.org: An Online Tool for Selecting Colour Schemes for Maps." *The Cartographic Journal* 40, no. 1 (): 27–37. Visited on 01/18/2014. doi:10.1179/000870403235002042. (Cit. on p. 59).
- Hendry, D. G., and T. R. G. Green. 1994. "Creating, comprehending and explaining spreadsheets: a cognitive interpretation of what discretionary users think of the spreadsheet model." *International Journal of Human-Computer Studies* 40, no. 6 (): 1033–1065. Visited on 09/13/2015. doi:10.1006/ijhc.1994.1047. (Cit. on p. 66).
- Hill, William C., James D. Hollan, Dave Wroblewski, and Tim McCandless. 1992. "Edit Wear and Read Wear." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 3–9. CHI '92. New York, NY, USA: ACM. Visited on 09/20/2014. doi:10.1145/142750.142751. (Cit. on p. 114).

- Hoarau, Raphaël, and Stéphane Conversy. 2012. "Augmenting the Scope of Interactions with Implicit and Explicit Graphical Structures." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 1937–1946*. CHI '12. New York, NY, USA: ACM. Visited on 09/20/2016. doi:[10 . 1145 / 2207676 . 2208337](https://doi.org/10.1145/2207676.2208337). (Cit. on pp. [56](#), [57](#), [61](#)).
- Hsi, Idris, and Colin Potts. 2000. "Studying the evolution and enhancement of software features." In *Software Maintenance, 2000. Proceedings. International Conference on*, 143–151. IEEE. Visited on 05/19/2015. (Cit. on p. [63](#)).
- Itten, Johannes. 1970. *The Elements of Color: A Treatise on the Color System of Johannes Itten Based on His Book the Art of Color*. 1 edition. Ed. by Faber Birren. Trans. by Ernst Van Hagen. New York: John Wiley & Sons. (Cit. on pp. [24](#), [25](#), [27](#), [31](#), [32](#)).
- . 1973. *The art of color: the subjective experience and objective rationale of color*. New York: John Wiley. (Cit. on p. [31](#)).
- Jacobs, Charles, Wilmot Li, Evan Schrier, David Barger, and David Salesin. 2003. "Adaptive grid-based document layout," 838. ACM Press. Visited on 07/31/2015. doi:[10.1145/1201775.882353](https://doi.org/10.1145/1201775.882353). (Cit. on p. [51](#)).
- Jahanian, Ali, Jerry Liu, Qian Lin, Daniel Tretter, Eamonn O'Brien-Strain, Seungyon Claire Lee, Nic Lyons, and Jan Allebach. 2013. "Recommendation system for automatic design of magazine covers." In *Proceedings of the 2013 international conference on Intelligent user interfaces*, 95–106. ACM. Visited on 03/05/2015. (Cit. on p. [50](#)).
- Jalal, Ghita, Nolwenn Maudet, and Wendy E. Mackay. 2015. "Color Portraits: From Color Picking to Interacting with Color." In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 4207–4216. CHI '15. New York, NY, USA: ACM. Visited on 09/15/2015. doi:[10 . 1145 / 2702123 . 2702173](https://doi.org/10.1145/2702123.2702173). (Cit. on pp. [6](#), [87](#), [100](#)).
- Javed, Waqas, Niklas Elmquist, Ji Soo Yi, and others. 2011. "Direct manipulation through surrogate objects." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 627–636. ACM. Visited on 05/24/2016. (Cit. on pp. [52](#), [53](#), [61](#)).
- Johnson, Jeff, Teresa L. Roberts, William Verplank, David C. Smith, Charles H. Irby, Marian Beard, and Kevin Mackey. 1989. "The Xerox Star: A Retrospective." *Computer* (Los Alamitos, CA, USA) 22, no. 9 (): 11–26, 28–29. doi:[10.1109/2.35211](https://doi.org/10.1109/2.35211). (Cit. on pp. [1](#), [43](#), [46](#), [48](#), [49](#), [51](#), [56](#), [61](#), [64](#), [74](#), [80](#), [81](#)).

- Johnson, Timothy E. 1963. "Sketchpad III: a computer program for drawing in three dimensions." In *Proceedings of the May 21-23, 1963, spring joint computer conference*, 347-353. ACM. Visited on 07/27/2016. (Cit. on pp. 44, 47).
- Kaufman, Leah, and Brad Weed. 1998. "Too much of a good thing?: identifying and resolving bloat in the user interface." In *CHI 98 Conference Summary on Human Factors in Computing Systems*, 207-208. ACM. Visited on 02/01/2015. (Cit. on p. 63).
- Kay, A., and A. Goldberg. 1977. "Personal Dynamic Media." *Computer* 10, no. 3 (): 31-41. Visited on 07/28/2016. doi:[10.1109/C-M.1977.217672](https://doi.org/10.1109/C-M.1977.217672). (Cit. on pp. 45-47).
- Kay, Alan C. 1972. "A Personal Computer for Children of All Ages." In *Proceedings of the ACM Annual Conference - Volume 1*. ACM '72. New York, NY, USA: ACM. Visited on 07/28/2016. doi:[10.1145/800193.1971922](https://doi.org/10.1145/800193.1971922). (Cit. on p. 45).
- Kazi, Rubaiat Habib, Fanny Chevalier, Tovi Grossman, Shengdong Zhao, and George Fitzmaurice. 2014a. "DRACO: Sketching Animated Drawings with Kinetic Textures." In *ACM SIGGRAPH 2014 Posters*, 5:1-5:1. SIGGRAPH '14. New York, NY, USA: ACM. Visited on 09/20/2016. doi:[10.1145/2614217.2614221](https://doi.org/10.1145/2614217.2614221). (Cit. on p. 57).
- Kazi, Rubaiat Habib, Fanny Chevalier, Tovi Grossman, and George Fitzmaurice. 2014b. "Kitty: sketching dynamic and interactive illustrations," 395-405. ACM Press. Visited on 06/08/2016. doi:[10.1145/2642918.2647375](https://doi.org/10.1145/2642918.2647375). (Cit. on p. 57).
- King, Timothy D. 2005. "Human color perception, cognition, and culture: why red is always red." In *Electronic Imaging 2005*, 234-242. International Society for Optics / Photonics. Visited on 02/10/2014. (Cit. on p. 23).
- Kirton, Travis. 2011. "Artistic canvases for gestural and non-linear typography." In *Proceedings of the International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging*, 63-70. ACM. Visited on 09/21/2016. (Cit. on p. 55).
- Kubat, Rony, Daniel Mirman, and Deb Roy. 2009. "Semantic context effects on color categorization." Visited on 07/01/2014. (Cit. on p. 24).
- Kuhna, Mikko, Ida-Maria Kivelä, and Pirkko Oittinen. 2012. "Semi-automated magazine layout using content-based image features," 379. ACM Press. Visited on 11/12/2014. doi:[10.1145/2393347.2393403](https://doi.org/10.1145/2393347.2393403). (Cit. on p. 50).
- Lakoff, George, and Mark Johnson. 1980. *Metaphors we live by*. Chicago: University of Chicago Press. (Cit. on p. 48).

- Lauer, David A., and Stephen Pentak. 2012. *Design basics*. 8th ed. OCLC: ocn651909138. Boston, MA: Wadsworth, Cengage Learning. (Cit. on pp. 19–21, 38).
- Lee, Brian, Sivil Srivastava, Ranjitha Kumar, Ronen Brafman, and Scott R. Klemmer. 2010. "Designing with interactive example galleries." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2257–2266. ACM. Visited on 06/29/2015. (Cit. on p. 60).
- Liu, Jianguy, Jian Sun, and Heung-Yeung Shum. 2009. "Paint selection." *ACM Transactions on Graphics* 28, no. 3 (): 1. Visited on 05/24/2016. doi:10.1145/1531326.1531375. (Cit. on p. 58).
- Ludolph, Frank, and Roderick Perkins. 1998. "The Lisa user interface." In *CHI 98 Conference Summary on Human Factors in Computing Systems*, 18–19. ACM. Visited on 08/08/2016. (Cit. on pp. 1, 48).
- Mackay, Wendy E. 2002a. "Which interaction technique works when?: floating palettes, marking menus and toolglasses support different task strategies." In *Proceedings of the Working Conference on Advanced Visual Interfaces*, 203–208. Bibtex: mackay_which_2002. Visited on 04/05/2013. (Cit. on p. 130).
- Mackay, Wendy E., and Anne-Laure Fayard. 1997. "HCI, natural science and design: a framework for triangulation across disciplines." In *Proceedings of the 2nd conference on Designing interactive systems: processes, practices, methods, and techniques*, 223–234. ACM. Visited on 02/10/2014. (Cit. on p. 4).
- Mackay, Wendy. 2002b. *Using Video to Support Interaction Design*. Visited on 09/09/2016. (Cit. on pp. 4, 5).
- . 2014. *Structured Observation to Support Interaction Design*. (Cit. on pp. 4, 6).
- McGrenere, Joanna, and Gale Moore. 2000. "Are we all in the same" bloat"?" In *Graphics interface, 2000:187–196*. Visited on 02/01/2015. (Cit. on pp. 63, 67).
- Meier, Barbara J. 1988. "ACE: a color expert system for user interface design." In *Proceedings of the 1st annual ACM SIGGRAPH symposium on User Interface Software*, 117–128. ACM. Visited on 06/26/2014. (Cit. on pp. 50, 85).
- Meier, Barbara J., Anne Morgan Spalter, and David B. Karelitz. 2004. "Interactive color palette tools." *Computer Graphics and Applications, IEEE* 24 (3): 64–72. Visited on 01/18/2014. (Cit. on p. 59).
- Montazeri, S., R. Gonzalez, C. Yoon, P. Y. Papalambros, and others. 2012. "Color, cognition, and recycling: How the design of everyday objects prompt behavior change." In *DS 70: Proceedings of DESIGN 2012, the 12th International Design Conference, Dubrovnik, Croatia*. Visited on 08/17/2016. (Cit. on p. 24).

- Moulder, Peter, and Kim Marriott. 2012. "Learning how to trade off aesthetic criteria in layout," 33. ACM Press. Visited on 07/31/2015. doi:[10.1145/2361354.2361361](https://doi.org/10.1145/2361354.2361361). (Cit. on p. 60).
- Munsell, A. H. 1905. *A Color Notation*. G. H. Ellis Company. (Cit. on pp. 27, 28).
- . 1915. *Atlas of the Munsell color system*. [Malden, Mass., Wadsworth, Howland & Co., inc., Printers]. Visited on 08/23/2016. (Cit. on p. 28).
- Munsell, A.H. 1929. *Munsell book of color: defining, explaining, and illustrating the fundamental characteristics of color*. v. 1. Munsell Color Co. (Cit. on p. 28).
- Myers, Brad A. 1998. "A brief history of human-computer interaction technology." *interactions* 5 (2): 44–54. Visited on 09/14/2015. (Cit. on pp. 58, 63).
- Myers, Brad A., Ashley Lai, Tam Minh Le, YoungSeok Yoon, Andrew Faulring, and Joel Brandt. 2015. "Selective Undo Support for Painting Applications," 4227–4236. ACM Press. Visited on 07/06/2016. doi:[10.1145/2702123.2702543](https://doi.org/10.1145/2702123.2702543). (Cit. on p. 58).
- Napier, H. Albert, Richard R. Batsell, David M. Lane, and Norman S. Guadagno. 1992. "Knowledge of command usage in a spreadsheet program." *ACM SIGMIS Database* 23 (1): 13–21. Visited on 06/26/2016. (Cit. on p. 67).
- Nardi, Bonnie A., and James R. Miller. 1991. "Twinkling lights and nested loops: distributed problem solving and spreadsheet development." *International Journal of Man-Machine Studies* 34 (2): 161–184. Visited on 01/22/2014. (Cit. on p. 66).
- O'Donovan, Peter, Aseem Agarwala, and Aaron Hertzmann. 2015. "DesignScape: Design with Interactive Layout Suggestions," 1221–1224. ACM Press. Visited on 05/05/2015. doi:[10.1145/2702123.2702149](https://doi.org/10.1145/2702123.2702149). (Cit. on pp. 50, 51).
- OKI Data. 2016. *Printer Color Matching Made Easy*. Visited on 08/24/2016. (Cit. on p. 30).
- Pang, Wai-Man. 2010. "An intuitive texture picker." In *Proceedings of the 15th international conference on Intelligent user interfaces*, 365–368. ACM. Visited on 09/13/2014. (Cit. on p. 116).
- Piccoli, Ricardo Farias Bidart, Rodrigo Chamun, Nicole Carrion Cogo, João Batista Souza de Oliveira, and Isabel Harb Manssour. 2011. "A novel physics-based interaction model for free document layout." In *Proceedings of the 11th ACM symposium on Document engineering*, 153–162. ACM. Visited on 06/29/2015. (Cit. on p. 50).
- Pomerantz, James R., and Anna I. Cragin. 2015. "Emergent features and feature combination." *The Oxford handbook of perceptual organization*:88–107. Visited on 09/07/2016. (Cit. on pp. 11, 16).

- Pomerantz, James R, and Michael Kubovy. 1986. "Theoretical approaches to perceptual organization: Simplicity and likelihood principles." *Organization* 36 (3). (Cit. on pp. 15, 17).
- Pouli, Tania, and Erik Reinhard. 2010. "Progressive histogram reshaping for creative color transfer and tone reproduction." In *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*, 81–90. ACM. Visited on 06/11/2014. (Cit. on p. 59).
- Reis, Joaquim. 2008. "Using rules for creativity in visual composition." In *Proceedings of the 26th annual ACM international conference on Design of communication*, 207–214. ACM. Visited on 03/05/2015. (Cit. on p. 50).
- Russel, Bertrand. 1948. *Human Knowledge - Its Scope and Limits*. Simon and Schuster. 1948. Simon & Schuster. (Cit. on p. 23).
- Sajaniemi, Jorma, and Jari Pekkanen. 1988. "An empirical analysis of spreadsheet calculation." *Software: Practice and Experience* 18, no. 6 (): 583–596. Visited on 06/26/2016. doi:[10.1002/spe.4380180607](https://doi.org/10.1002/spe.4380180607). (Cit. on p. 66).
- Samara, Timothy. 2007. *Design Elements: A Graphic Style Manual*. First Edition edition. Gloucester, Mass: Rockport Publishers. (Cit. on pp. 18, 19, 22, 33, 34).
- Schon, Donald A. 1984. *The Reflective Practitioner: How Professionals Think In Action*. 1 edition. New York: Basic Books. (Cit. on pp. 12, 39).
- . 1992. "Designing as reflective conversation with the materials of a design situation." *Research in Engineering Design* 3 (3): 131–147. Visited on 02/23/2016. (Cit. on pp. 39, 40).
- Schrier, Evan, Mira Dontcheva, Charles Jacobs, Geraldine Wade, and David Salesin. 2008. "Adaptive layout for dynamically aggregated documents," 99. ACM Press. Visited on 08/03/2015. doi:[10.1145/1378773.1378787](https://doi.org/10.1145/1378773.1378787). (Cit. on p. 50).
- Shneiderman, Ben. 1983. "Direct manipulation: a step beyond programming languages." *Sparks of innovation in human-computer interaction* 17:1993. (Cit. on p. 47).
- Sinha, Nishant, and Rezwana Karim. 2015. "Responsive Designs in a Snap." In *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*, 544–554. ESEC/FSE 2015. New York, NY, USA: ACM. Visited on 09/15/2016. doi:[10.1145/2786805.2786808](https://doi.org/10.1145/2786805.2786808). (Cit. on p. 60).
- Sloane, Patricia. 1989. *The Visual Nature of Color*. First Edition edition. New York: Design Pr. (Cit. on pp. 22–24).

- Smith, David C., Frank E. Ludolph, and Charles H. Irby. 1985. "The desktop metaphor as an approach to user interface design (panel discussion)," 548–549. ACM Press. Visited on 08/05/2016. doi:[10.1145/320435.320594](https://doi.org/10.1145/320435.320594). (Cit. on p. 48).
- St Amant, Robert, and Thomas E. Horton. 2002. "Characterizing tool use in an interactive drawing environment." In *Proceedings of the 2nd international symposium on Smart graphics*, 86–93. ACM. Visited on 06/08/2016. (Cit. on p. 51).
- Staples, Loretta. 2000. "Typography and the Screen: A Technical Chronology of Digital Typography, 1984–1997." *Design Issues* 16, no. 3 (): 19–34. Visited on 08/30/2016. doi:[10.1162/07479360052053306](https://doi.org/10.1162/07479360052053306). (Cit. on p. 35).
- Suchman, Lucille Alice. 2007. *Human-machine reconfigurations: plans and situated actions*. OCLC: 64592145. Cambridge; New York: Cambridge University Press. (Cit. on pp. 40, 41).
- Sukale, Ryan, Olesia Koval, and Stephen Voida. 2014. "The Proxemic Web: Designing for Proxemic Interactions with Responsive Web Design." In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, 171–174. UbiComp '14 Adjunct. New York, NY, USA: ACM. Visited on 06/10/2015. doi:[10.1145/2638728.2638768](https://doi.org/10.1145/2638728.2638768). (Cit. on p. 50).
- Sutherland, Ivan E. 1964. "Sketch pad a man-machine graphical communication system." In *Proceedings of the SHARE design automation workshop*, 6–329. ACM. Visited on 07/27/2016. (Cit. on pp. 43, 44).
- Terry, Michael, and Elizabeth D. Mynatt. 2002a. "Recognizing Creative Needs in User Interface Design." In *Proceedings of the 4th Conference on Creativity & Cognition*, 38–44. C&C '02. New York, NY, USA: ACM. Visited on 02/23/2015. doi:[10.1145/581710.581718](https://doi.org/10.1145/581710.581718). (Cit. on p. 54).
- . 2002b. "Side views: persistent, on-demand previews for open-ended tasks," 71. ACM Press. Visited on 04/08/2015. doi:[10.1145/571985.571996](https://doi.org/10.1145/571985.571996). (Cit. on pp. 54, 60, 65).
- Terry, Michael, Elizabeth D. Mynatt, Kumiyo Nakakoji, and Yasuhiro Yamamoto. 2004. "Variation in element and action: supporting simultaneous development of alternative solutions." In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 711–718. ACM. Visited on 05/21/2014. (Cit. on pp. 54, 55, 60).
- Tristram, Claire. 2001. *The Next Computer Interface*. Visited on 08/01/2016. (Cit. on p. 49).
- Tschichold, Jan, Robin Kinross, and Richard Hendel. 2006. *The New Typography*. 1 edition. Trans. by Ruari McLean. Berkeley, Calif.; London: University of California Press. (Cit. on p. 34).

- Visser, Willemien. 2010. "Schön: Design as a reflective practice." *Collection*, no. 2:21–25. Visited on 09/01/2016. (Cit. on pp. 39, 65, 67).
- Wagemans, Johan, Jacob Feldman, Sergei Gepshtein, Ruth Kimchi, James R. Pomerantz, Peter A. van der Helm, and Cees van Leeuwen. 2012. "A century of Gestalt psychology in visual perception: II. Conceptual and theoretical foundations." *Psychological Bulletin* 138 (6): 1218–1252. Visited on 09/02/2016. doi:10.1037/a0029334. (Cit. on p. 16).
- Wandell, Brian A. 1995. *Foundations of Vision*. 1 edition. Sunderland, Mass: Sinauer Associates Inc. (Cit. on pp. 22, 23).
- Wang, Lujin, Joachim Giesen, Kevin T. McDonnell, Peter Zolliker, and Klaus Mueller. 2008. "Color design for illustrative visualization." *Visualization and Computer Graphics, IEEE Transactions on* 14 (6): 1739–1754. Visited on 01/18/2014. (Cit. on p. 50).
- Williams, Gregg. 1984. "Apple Macintosh computer." *Byte* 9 (2): 30–31. Visited on 07/04/2016. (Cit. on pp. 1, 49, 63).
- Williamson, Jack H. 1986. "The Grid: History, Use, and Meaning." *Design Issues* 3 (2): 15–30. Visited on 08/25/2016. doi:10.2307/1511481. (Cit. on p. 38).
- Wolfe, Jeremy, and Lynn Robertson. 2012. *From Perception to Consciousness: Searching with Anne Treisman*. Google-Books-ID: f_NgRX1DJ7IC. OUP USA. (Cit. on p. 15).
- Xia, Haijun, Bruno Araujo, Tovi Grossman, and Daniel Wigdor. 2016. "Object-Oriented Drawing," 4610–4621. ACM Press. Visited on 05/19/2016. doi:10.1145/2858036.2858075. (Cit. on pp. 53, 54, 61, 130).
- Xu, Pengfei, Hongbo Fu, Chiew-Lan Tai, and Takeo Igarashi. 2015. "GACA: Group-Aware Command-based Arrangement of Graphic Elements," 2787–2795. ACM Press. Visited on 06/29/2015. doi:10.1145/2702123.2702198. (Cit. on p. 60).
- Zaman, Loutfouz, Wolfgang Stuerzlinger, Christian Neugebauer, Rob Woodbury, Maher Elkhaldi, Naghmi Shireen, and Michael Terry. 2015. "GEM-NI: A System for Creating and Managing Alternatives In Generative Design," 1201–1210. ACM Press. Visited on 05/05/2015. doi:10.1145/2702123.2702398. (Cit. on p. 60).

Titre : Reified properties for visual composition

Mots-clés : composition, visual properties, creativity, design, color, layout

In this dissertation, I argue that visual composition tools should reify visual properties— that is, create them into first-class, interactive objects that designers can manipulate, directly in the document. Artists and designers use visual properties such as color, typography, size, and position to create novel composition concepts. Most visual composition tools treat graphical elements as objects— but not their visual properties. The latter are simply attributes of graphical elements, usually accessible on demand, through property sheets or dialog boxes, rather than as independent interactive objects.

I begin with an introduction, where I outline the dissertation and summarise the main contributions. Then, I present a conceptual overview, where I argue that perception, expertise, theories and existing representations of visual properties affect how designers manipulate them in their work. In the related work section, I identify the main approaches visual composition tools propose to manipulate visual properties. I explain the advantages and limitations of these approaches.

I present a comparative structured observation study (Chapter 4) where graphic designers perform visual composition tasks in Adobe Illustrator. Designers preferred tools that provide direct access to visual properties. They use these visual properties to complete visual composition tasks such as alignment and distribution. I then present two interview studies (Chapter 5, Chapter 6) that examined artists' and designers' practices as they manipulate color and create lay-

outs in their projects. Artists and designers create personal color representations. They manipulate each color in the context of its surrounding graphical elements, and combine it with other visual properties such as texture. As they create their layouts, designers establish links among visual properties such as size, position, and layering of graphical elements. They define rules for how these properties change in space, across instances of the same composition, or in time, across related compositions.

Based on these observations, two groups of prototypes (Chapter 7) demonstrate how we can reify visual properties into first-class graphical objects. Color Partner supports creating personal representations of the color manipulation process. Color Revealer permits creating informative color representations of how a process evolves. Palette Explorer supports manipulating visual properties of color swatches, such as size, position and layering, in the context of surrounding colors. Color Composer supports manipulating the combination of colors and textures. Linkify supports linking visual properties directly in the document and defining rules for how these links evolve as the content changes. Contextify supports creating rules that define how visual properties change across instances of a composition. In two follow up studies, designers explain how they intend to use these tools in their current projects.

I conclude with a summary of the contributions (Chapter 8) and a discussions of the limitations and future work.

Titre : Réification des propriétés visuelles pour les tâches de composition

Mots-clés : composition, propriétés visuelles, créativité, design, couleur, mise en page

Les graphistes utilisent des propriétés visuelles comme la couleur, la police de caractères typographiques, la taille ou la position pour construire et communiquer leurs concepts visuels. La plupart des outils numériques de composition considèrent les propriétés visuelles comme simples attributs. Ces propriétés sont accessibles, quand l'objet est sélectionné, via des boîtes de dialogue, sans être considérées comme objets indépendants dans l'interface. Cette thèse porte sur la réification des propriétés visuelles des objets graphiques pour les tâches de composition. Je soutiens l'importance de créer des outils de composition qui permettent de manipuler les propriétés visuelles directement, sans passer par des menus ou des boîtes de dialogue.

La manipulation des propriétés visuelles dépend de la perception des propriétés, de l'expertise de l'utilisateur et de leur représentations sur l'interface. Elle dépend également des outils existants pour manipuler les propriétés visuelles dans le contexte de travail. Les outils numériques de composition utilisent principalement des boîtes de dialogue. Certains travaux de recherche soutiennent l'importance de faire évoluer les boîtes de dialogues, et proposent des boîtes de dialogues plus interactives. D'autres travaux proposent de nouvelles manières de représenter et de manipuler les propriétés visuelles.

Les graphistes préfèrent les outils qui agissent directement sur les propriétés visuelles. Ils adaptent ces propriétés, pour les utiliser comme outils leur permettant de compléter des tâches comme l'alignement et l'espacement.

Les artistes et les designers travaillent en développant leurs propres représentations des espaces de couleur. Ils manipulent chaque couleur, dans le contexte des autres couleurs utilisées dans le même document. Leurs choix de couleur dépendent également d'autres propriétés visuelles comme la texture, la matière, ou la luminosité. En créant leurs mise en page, les graphistes établissent des liens entre les propriétés spatiales comme la position, la taille, et la superposition des éléments graphiques. Ils créent des règles pour définir comment ces liens évoluent dans un même document, ou entre différents documents.

Deux groupes d'outils proposent des exemples de la réification des propriétés visuelles pour des tâches de composition. Deux études avec des designers présentent différentes utilisations envisagées pour ces outils. Chaque utilisateur présente des exemples de projets spécifiques dans lesquels ces outils faciliteront leurs tâches de composition, et comment ils leur permettront d'obtenir les rendus recherchés.

Titre : Réification des propriétés visuelles pour les tâches de composition

Mots-clés : composition, propriétés visuelles, créativité, design, couleur, mise en page

Les graphistes utilisent des propriétés visuelles comme la couleur, la police de caractères typographiques, la taille ou la position pour construire et communiquer leurs concepts visuels. La plupart des outils numériques de composition considèrent les propriétés visuelles comme simples attributs. Ces propriétés sont accessibles, quand l'objet est sélectionné, via des boîtes de dialogue, sans être considérées comme objets indépendants dans l'interface. Cette thèse porte sur la réification des propriétés visuelles des objets graphiques pour les tâches de composition. Je soutiens l'importance de créer des outils de composition qui permettent de manipuler les propriétés visuelles directement, sans passer par des menus ou des boîtes de dialogue.

L'introduction résume les contributions de cette thèse et présente les méthodes d'observation, d'interviews, et d'analyse de données utilisées pour la conduite des études présentées dans les chapitres suivants. Je poursuis par la présentation des facteurs qui conditionnent la manipulation des propriétés visuelles dans les tâches de composition. Cette manipulation dépend de la perception des propriétés, de l'expertise de l'utilisateur et de leur représentations. Elle dépend également des outils existants pour manipuler les propriétés visuelles dans le contexte de travail. L'état de l'art présente les approches principales adoptées par les outils graphiques pour la manipulation des propriétés visuelles. Les outils de composition commerciaux utilisent principalement des boîtes de dialogue. Certains travaux de recherche soutiennent l'importance de faire évoluer les boîtes de dialogues, et proposent des boîtes de dialogues plus interactives. D'autres travaux de recherche proposent de nouvelles manières de représenter et de manipuler les propriétés visuelles.

Je présente une étude (Chapter 4) dans laquelle des graphistes professionnels recréent deux compositions dans une application dédiée à ce type de tâches. Cette étude révèle que les graphistes préfèrent les outils qui agissent directement sur les propriétés visuelles. Ils adaptent ces propriétés, pour les utiliser comme outils leur permettant de compléter des tâches comme

l'alignement et l'espacement.

Je présente ensuite deux études (chapitres 5 et 6) portant sur les pratiques de manipulation de la couleur, et de mise en page de documents chez les artistes et les designers. Ces praticiens travaillent en développant leurs propres représentations des espaces de couleur. Ils manipulent chaque couleur, dans le contexte des autres couleurs utilisées dans le même document. Leurs choix de couleur dépendent également d'autres propriétés visuelles comme la texture, la matière, ou la luminosité. En créant leur mise en page, les graphistes établissent des liens entre les propriétés spatiales comme la position, la taille, et la superposition des éléments graphiques. Ils créent des règles pour définir comment ces liens évoluent dans un même document, ou entre différents documents.

Deux groupes d'outils (chapitre 7) proposent des exemples de la réification des propriétés visuelles pour des tâches de composition. Color Partner permet de créer des représentations personnelles de couleurs. Color Revealer utilise la couleur pour créer une représentation informative d'un processus de travail. Color Compositor permet de manipuler des combinaisons de couleurs et de textures, et de tester ces compositions directement sur l'outil. Palette Explorer permet de manipuler les propriétés visuelles de carrés de couleur, afin de supporter le choix d'une couleur dans le contexte des autres couleurs de la palette. Linkify permet de créer des liens entre les propriétés spatiales des éléments graphiques. Contextify permet de définir des règles pour gérer l'évolution des propriétés visuelles dans une composition. Deux études avec des designers présentent différentes utilisations envisagées pour ces outils. Chaque utilisateur présente des exemples de projets spécifiques dans lesquels ces outils faciliteront leurs tâches de composition, et comment ils leur permettront d'obtenir les rendus recherchés.

En conclusion (chapitre 8), je résume les principales contributions de cette thèse, et les limitations et extensions possible du travail présenté.

