# UNDERSTANDING INTERACTION WITH COMMAND INTERFACES

Philip Quinn
ppq10@cosc.canterbury.ac.nz

Department of Computer Science and Software Engineering
University of Canterbury
Christchurch, New Zealand

**Supervisor**: Associate Professor Andy Cockburn
andy@cosc.canterbury.ac.nz

8 November 2007

## ABSTRACT

Developing models for describing the influences of interfaces on user interaction is a continuing goal of human-computer interaction research. Such models are developed through development of understanding and insight using the scientific method of observation and evaluation. We present a classification framework as a foundation for developing these models specifically for command interfaces. This classification describes the components of interaction around four top-level categories: (1) organisation, (2) navigation, (3) interaction, and (4) presentation—as a set of theories and design considerations. We then develop the aspect of navigation, building a set of principles that describe the factors that influence it. Finally, we describe an empirical evaluation of one of these principles—a potential model for landmarking interfaces, that describes a logarithmic relationship between the number of visible items and the number of presented landmarks—and found it to hold true for the evaluation interface.

## ACKNOWLEDGEMENTS

The author would like to thank his supervisor, Andy Cockburn, for his guidance, inspiration, and sanity throughout this project. He would also like to thank his family, and the staff members of the Computer Science department for their support throughout the year. All of the participants who volunteered their time to take part in the evaluation also deserve thanks for their time and valuable feedback. Finally, the author would like to thank his fellow Honours students of 2007 for making this a fantastic year.

# Contents

CHAPTER 1

# Introduction

Command interfaces are the front-line of user interaction when working with applications; they expose the tools to manipulate, navigate, and control the presentation of, a user's documents and data. The need for command interfaces to be simple to use and understandable by users, whilst being powerful and flexible to their needs, has been the subject of considerable research.

The fruit of this research has been the evolution of command interfaces as they struggle to maintain this balance. For example, text-editors have evolved from the powerful but mentally demanding interfaces of `vi`[1] and `emacs`[2], to the simpler but immediately understandable editors such as UltraEdit[3] and gedit[4]. This evolution has been prompted by the need to consider user interaction on a range of levels—such as the need for visual feedback and the paradigms of graphical user interfaces.

Evaluation of command interfaces typically takes the form of empirical examination of a novel design, set against existing interfaces. Although this type of evaluation has advanced the look and feel of user interfaces, it does little to develop our understanding of the fundamental influences of user performance. In order to develop insight into these influences, we need to establish what the components of command interfaces are, and the theory that models or explains interaction with them.

In this report, we develop a classification of command interfaces—defining a set of top-level categories that describe the theory and design considerations applicable to all command interfaces: (1) organisation, (2) navigation, (3) interaction, and (4) presentation.

This classification is a framework for building new principles and models for describing interaction with user interfaces. We explore the set of principles surrounding the aspects of command interface navigation—the components of, and influences on, the "navigability" and user efficiency in command interfaces.

[1] `http://www.vim.org/`
[2] `http://www.gnu.org/software/emacs/`
[3] `http://www.ultraedit.com/`
[4] `http://www.gnome.org/projects/gedit/`

Finally, we take the principle of landmarks—describing the number of landmarks to present to the user in a navigation view, in terms of the number of currently visible items—and conduct an empirical evaluation of it in order to establish its strength and validity.

The structure of this report is as follows: chapter 2 explores the related work and builds the framework for our classification hierarchy; chapter 3 takes the navigation branch of the classification hierarchy and uses it to build a set of principles that influence navigation in command interfaces; chapter 4 details an empirical evaluation of one of these principles; and a discussion of the work is presented in chapter 5.

CHAPTER 2

# Classification of Interface Theory

Command interfaces have taken many different forms as they continue to evolve. Extensive research across diverse disciplines has developed formal theories on the aspects of user interface design. Although designers are not consistently aware of formal theory, these theories are important in order to understand user behaviour and are useful in predicting and modelling user interaction and performance. Classifications, such as those by Bier *et al.* (1994) of transparent tools, Jul and Furnas (1997) of navigation issues in electronic worlds, Chignell (1990) of user interface terminology, and Beaudouin-Lafon (2000) of interaction techniques, allow for greater insight into their respective areas.

This chapter presents the foundation of a classification covering the breadth of theory relevant to command interfaces in consideration of their implications for design. Unifying these theories allows for greater insight to be gained into the reasoning and methodology behind interface design decisions, the interaction between these theories, and the development of models for user interaction.

## 2.1 CLASSIFICATION DESIGN

The classification has been designed around four top-level categories which encompass the major aspects of command interface design. Within each category, a hierarchical set of considerations are presented that describe and model the influences on user interaction. An overview of the classification hierarchy is shown in figure 2.1 and reviewed in section 2.6. These categories are not isolated groups; the issues raised by one category influence aspects of another—for example, a chosen organisation structure influences the navigation requirements for the structure, which subsequently has consequences for the chosen interaction methods and interface presentation.

The top-level categories of the classification hierarchy are:

Organisation The organisation of a command interface considers how designers place commands into a particular structure or conceptual organisational space (such as a hierarchal tree or connected graph), and the properties of those structures that influence user interaction and efficiency. It is important to note that this

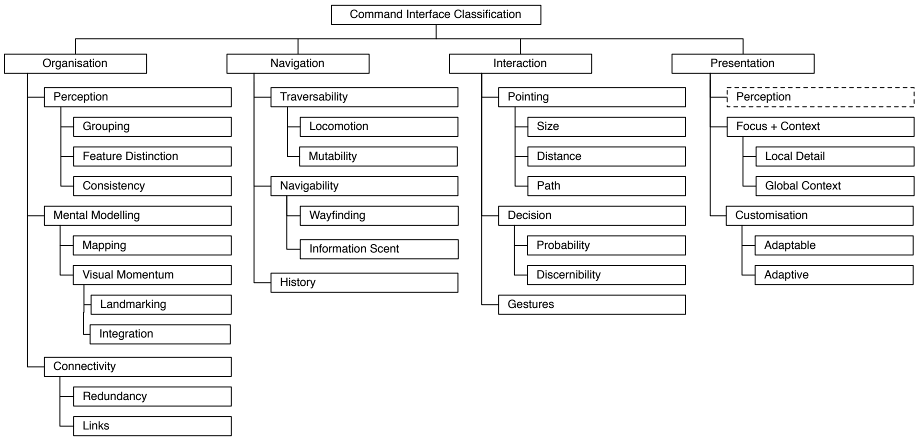



*Figure 2.1*: An overview of the classification hierarchy.

organisation is independent from the visual presentation of the user interface.

**Navigation** A user's ability to traverse through a command structure is classified as the navigation. These are the properties that allow traversal through the interface's states and data. This will typically involve enabling the dynamic manipulation of a static organisational structure.

**Interaction** The theories that model a user's interaction with the interface are presented in this category. This includes the process of issuing a command (such as decision and pointing times) and models of interaction techniques (such as gestures and steering).

**Presentation** The presentation of a command interface classifies the visual aspects presented to the user. In contrast to the previous four categories that classify the structural and interaction aspects, this category focuses on how these concepts are displayed to the user.

The following sections review the related work in each of these categories.

## 2.2 ORGANISATION

All command interfaces have an organisational structure to them, often in the form of a hierarchy. Even visually 'flat' interfaces—such as toolbars—exhibit hierarchical properties when considering issues such as each toolbar's priority in the interface (Au and Li 1998). Irrespective of the visual presentation of commands, their organisation as a hierarchy or set of relationships must be established to better allow users to formulate a mental model of the command set. The design of this structure is dependent on the context it is created for, but there are features of such structures that impact their effectiveness regardless of context.

The organisation is a static, well-defined structure that exists independent of the dynamic tools that allow for movement through it (discussed as the navigation aspects

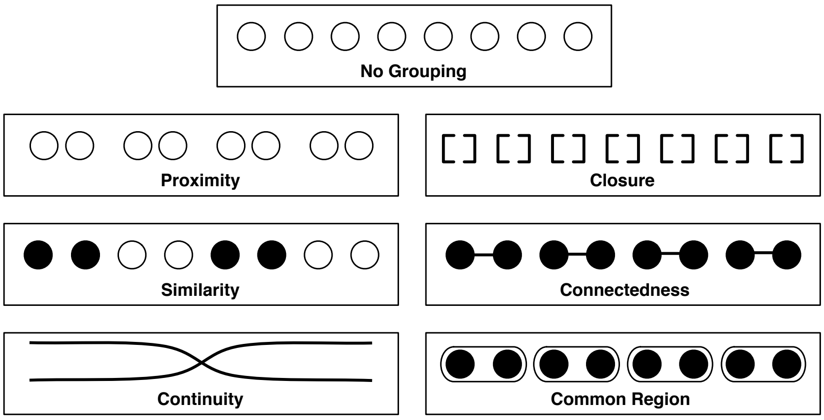



*Figure 2.2*: Gestalt psychology principles (reproduced from (Rock and Palmer 1990, p. 60)).

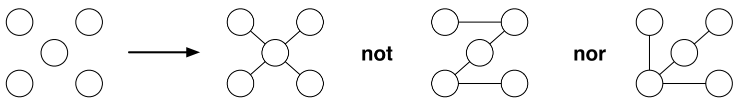



*Figure 2.3*: A task demonstrating the law of prägnanz: the visual field will be organised in the simplest or best way possible. (reproduced from (Pomerantz 1981, p. 161)).

in section 2.3). The issues raised here apply predominantly to organisational structures, but may also have implications for navigation (see chapter 3).

There are many factors when devising an effective organisational structure: the support for users to perceive distinctions between alternatives, the mental load required to understand and utilise a model of the organisational structure, and the assisting properties of the structure in aiding a user's attempts to find a target.

### 2.2.1 Perception

Gestalt psychology principles (Wertheimer 1958, Rock and Palmer 1990, Pomerantz 1981) are the seminal foundation for understanding people's perception of object grouping. Wertheimer (1958) outlined four factors that assist people in perceiving a set of objects as a group: (1) proximity, (2) similarity, (3) continuity, and (4) prägnanz (the minimum principle, demonstrated in figure 2.3). These factors have since been developed and expanded (Rock and Palmer 1990, Pomerantz 1981) to also include: (5) closure, (6) common region, and (7) connectedness (shown in figure 2.2). Empirical evaluations of these Gestalt factors reveal the prevailing strength of this pre-attentive grouping of objects: Beck (1966) observed a disconnect between how participants subjectively rated the similarity of items and the way 'Gestalt groups' of those items were formed. Coren and Girgus (1980) observed participants mentally distorting the spatial relationships between Gestalt groups of objects to support their maintenance of these perceptual groupings.

Treisman and Gelade's (1980) studies on visual search, and subsequent feature integration theory (FIT), revealed factors regarding sets of objects that allowed participants to perform pre-attentive, parallel examination of individual objects—significantly improving performance when later asked to acquire a target. Treisman and Gelade observed that participants initially tried to pre-attentively group similar objects based on their similarities and differences in several dimensions—such as colour, shape, orientation, and location. With enough distinction between the target(s) and the distractors, participants could perform rapid pre-attentive analysis; otherwise, participants fell back to a slower, serial examination of each item.

The Gestalt factors and Treisman and Gelade's feature integration theory influence how users can perceive relationships between interface items based on their differences and similarities in several 'perceptual primitives'. Although research has focused on the visual perception aspects of these theories, we believe they apply equally to the mental perception of a conceptual structure. For example, placing a set of interface commands on a single level of a hierarchy implies a certain organisational relationship between them. More complex relationships can be inferred by how commands are placed in the hierarchy—such as the connections between groups of commands, similarity in the organisation of groups, and the use of 'natural' organisation cues. These factors are important in assisting the user in developing a mental model without attentively examining every option in the hierarchy.

### 2.2.2 Mental Modeling

Gonzales (1994) presents a review of the theory for information organisation. Gonzales proposes the *mapping hypothesis*, which states that: "when the organisation of information in a user interface reflects the way users mentally organise the same information, user performance is enhanced" (p. 146). He reflects on Woods's (1984) theory of *visual momentum* that when supported by the effective organisation of a visual display (as described by several 'techniques'), the mental workload on the user is reduced, as they are able to infer connections and the relationships of the structure. Similarly, Norman *et al.*'s (1986) theory of *cognitive layout* argues that users will perform better when the physical organisation of interface elements accurately maps onto their mental representation of the interface, supported by further studies by Lee *et al.* (1984) and Chapanis and Lindenbaum (1959). Organisation structures that deny or hinder a connection between the physical model and a user's perceptual model run the risk of increasing user errors and frustration similar to that of the Stroop effect (Stroop 1935, Kahneman and Henik 1981).

Visual momentum is an important theory to consider when determining how to organise a command interface, and Woods presents several methods to assist visual momentum to avoid users "getting lost" on large computer displays and the "keyhole phenomena" in small displays. Woods identifies four techniques that assist in developing visual momentum and *spatial cognition*: (1) *long shot*, establishing views or overview of the structure; (2) *perceptual landmarks*, anchors as users transition between displays; (3) *display overlap*, assisting comprehension as users move between displays; and (4) *spatial representation*, the actual organisation topology or grouping, routes, and paths. The overall effect of these four techniques is the establishment of a mental "map" of the

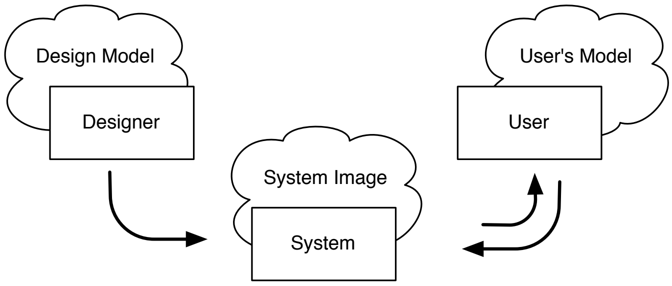



*Figure 2.4*: Norman's model of interaction (reproduced from (Norman 1988, p. 16)).

organisation that enables a visual momentum for the user when navigating—allowing the user to accurately induce relationships between points in the structure and reduce their mental workload. It has been shown through field studies that a lack of support of these techniques is a probable cause for disorientation in some user interfaces (de Alwis and Murphy 2006), and empirical evaluations of systems that support visual momentum techniques in comparison to those that do not, reveal significant improvement in efficiency for the systems supporting visual momentum (Tang 2001).

This mental modeling of an organisational structure can be related to Norman's model of interaction (Norman 1988, 1986, 1983) (reproduced in figure 2.4). A strong correlation between the designer's model and the user's model is needed for the user to easily use the device. However, the primary tool of the designer to convey their model is the system image that the user interacts with. "If the system image does not make the design model clear and consistent, then the user will end up with the wrong mental model" (Norman 1988, p. 16). It is important to consider the system image and its implications on the user's mental model of the system. Roske-Hofstrand and Paap (1986) proposed that for menus, organisation should be based on how users organise options, not what designers believe is logical.

### 2.2.3 Connectivity

Another important factor in the development of a user's mental model, and the utility of the organisational structure is how commands are connected together through hierarchies or links. Gonzales (1994) reflects on an evaluation by Roske-Hofstrand and Paap (1986) of a prototype menu system for an aeroplane cockpit with several levels of link redundancy (in terms of the number of links between nodes in the menu system). In empirical evaluation, users performed better with menu systems using a high level of redundancy than systems with lower levels, or without redundancy. The influence of redundancy in an organisational system would be to better facilitate the location of a single item relevant in many contexts and rapid access to items through reduced navigational requirements.

## 2.3 NAVIGATION

The navigation aspects of a command structure are the properties of the dynamic tools that allow for movement around a static organisational structure. There is a strong connection between organisation and navigation (further explored in chapter 3), but each has their own distinct properties, modelled by theory. As with organisation, these properties are considered independent of the visual presentation—they are properties of the navigational tools and their ability to support them.

### 2.3.1 Wayfinding and Locomotion

Jul and Furnas (1997) give an excellent review of a CHI workshop on the issues of navigation, including classifications of navigation aspects, tasks, and subtasks. Although they predominantly focus on navigation from a user's perspective in order to explore the influences on the designer's or system's perspective, they give insight into the characteristics of navigation interaction, arriving at four characteristics that contribute to design: (1) characteristics of the space to be navigated; (2) the task to be performed; (3) the strategy employed by the user; and (4) the user's existing knowledge of navigation in the system.

Nigay and Vernier (1998) reviewed the generalisation of all navigation tasks as consisting of *way-finding* + *locomotion* (Jul and Furnas 1997) and considered two axes for navigation tasks when analysing the design issues surrounding navigation in large information spaces—structural responsibility and target orientation—based on work by Waterworth and Chignell (1991). Structural responsibility refers to who has the responsibility for executing a task—either the user or the system; target orientation refers to the search of the interface (by the user) for a specific target, or the exploration of the interface without a definite goal. Nigay and Vernier proposed that in order to face the wide diversity of tasks that exist, multiple representational systems are needed to "make the information space perceivable by the user." These representational systems are the tools that allow users to view and navigate the information space and the representations of that information. Nigay and Vernier derived several recommendations for these systems: (1) the systems must be easy to change; (2) when moving between systems, temporal and visual continuity must be maintained to prevent the user from getting lost; (3) systems that present the focus of the user's attention must be precise and not distorted; (4) it should be possible to combine two systems to gain a wider view of the information space, throughout which, spatial continuity must be guaranteed; and (5) the navigational tools must be uniform for all representational systems.

### 2.3.2 View Navigation

Furnas (1997a, 1997b) presents a more specific model for navigation, comprising two parts—efficient view traversability (EVT) and view navigability (VN)—combined to form effective view navigation (EVN). Furnas considers navigation as the traversal across a directed graph of nodes (the "viewing graph", an interface to a "logical structure graph", demonstrated in figure 2.5). Properties of the viewing graph—such as the number of outgoing edges (with respect to the capabilities of the "local window"), or the shortest path between two nodes—influences the EVT of a structure. Properties that influence the ability to search for and find any other node from a particular node in the graph effect the VN of the structure.

Furnas makes two recommendations for EVT: (1) the number of outgoing edges in the viewing graph must be low compared to the size of the structure; and (2) the distance

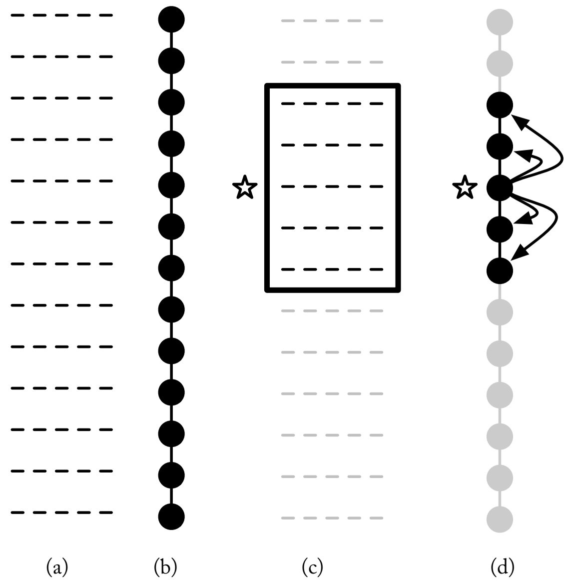



*Figure 2.5*: Furnas' EVT, showing (a) a list of items, (b) the logical graph of the list, (c) a local window view of the list focused at ⋆, and (d) the viewing graph for the local window in (c), showing the "out degree" (reproduced from (Furnas 1997b, p. 4)).

(in number of edges) between pairs of nodes in the viewing graph must be low compared to the size of the structure. He also presents several recommendations for improving the EVT of viewing graphs—such as "folding" them in multiple dimensions, "fisheye" sampling of the logical graph, and shortcut node augmentation of the viewing graph. A further three recommendations for VN are made: (1) all "outlink-info" (the information about where a particular node leads) must not be misleading; (2) every node must have good "residue" (information scent) at every other node; (3) outlink-info must be small.

There are strong connections between view navigation and "focus + context" interfaces that will be reviewed in section 2.5.2. The theory of EVN will be further reviewed in chapter 3.

### 2.3.3 Information Scent

The concepts of outlink-info and residue are explored as "information scent" (Pirolli 1997) by Pirolli and Card (1999) where the information foraging theory is used to understand the strategies of how users seek, gather, and consume information. *Information scent* is a set of cues found at one location to indicate what resides at another. Good information scent helps users better seek—or, navigate—by having a better indication about where they can locate the information they are looking for.

### 2.3.4 History

An aspect of navigation not commonly considered in most command structures is the revisitation of common commands in response to a user's needs for a given task or workflow. Greenberg and Witten (1993a) use empirical data to shown that a large percentage of commands issued on a system are repetitions of previous commands, with usage following a Zipfian (power-law) distribution (Zipf 1949) (further supported in studies by Findlater and McGrenere (2004) and Hanson *et al.* (1984)) with a very low command vocabulary growth rate. Greenberg and Witten (1993b) then present a set of recommendations for interfaces to support command reuse: (1) a user's previous activities should be available for recall; (2) activities should be grouped into high-level tasks and switching between tasks; and (3) allow end-user customisation of workspaces. Command reuse has been seen explored in several interfaces with positive results (Sears and Shneiderman 1994, Findlater and McGrenere 2004, Gajos *et al.* 2006), indicating that revisitation in navigation is an important aspect to consider.

### 2.3.5 Mutability

One of Greenberg and Witten's (1993b) recommendations is for "end-user customisation of workspaces", which they elaborate upon by proposing that "by merging a reuse facility with a customisable workspace … considerable power can be gained" (p. 395). They reflect on the work by MacLean *et al.* (1990) who discuss methods for creating "tailorable systems". Nigay and Vernier (1998) also include the ability to change representational systems as one of their recommendations. These interface adaptations tend to take the form of navigational shortcuts to provide greater accessibility to items—similar to a history mechanism, but spatially stable and with more control given to the user. Studies into adaptable interfaces (Stuerzlinger *et al.* 2006, Sears and Shneiderman 1994, Gajos *et al.* 2006, Findlater and McGrenere 2004) yield positive results when users understand the ability to personalise their interaction with the interface.

## 2.4 INTERACTION

Interaction covers the aspects relating to how a user directly manipulates and issues commands to the interface. Interaction with command interfaces can be modelled using a combination of foundational theoretical models—the "laws of action" (Zhai *et al.* 2004).

GOMS/KLM (Card *et al.* 1983) provide the seminal work for modeling tasks of user interaction. GOMS/KLM models user interaction to complete a task as a series of six operations: (1) pressing a key, (2) moving the pointer, (3) pressing or releasing a button, (4) hand movement from the mouse to keyboard (or vice versa), (5) mental thinking, and (6) system wait time. Although the model is limited in its application beyond expert tasks, GOMS and KLM models are useful tools for evaluating user processes and are open to augmentation by more precise models such as Fitts' law and the Hick-Hyman law.

### 2.4.1 Decision

The Hick-Hyman Law (Hick 1952, Hyman 1953) provides a model of the time taken for a user to make a decision amongst several choices. The law states that the time $T$ to choose an item is proportional to its information content (which is inversely proportional to its probability $p$), given by:

$$T = a + b \log_2(\frac{1}{p})$$

Or in the case for $n$ items $i$ with unequal probabilities $p_i$:

$$T = a + b \sum_{i}^{n} p_i \log_2(\frac{1}{p_i} + 1)$$

Where $a$ and $b$ are empirically derived constants. The Hick-Hyman law has been empirically shown to hold for navigation through on-screen hierarchies (Landauer and Nachbar 1985) and combined with Fitts' law to produce an accurate model of decision and pointing times in menus (Cockburn *et al.* 2007).

### 2.4.2 Pointing

Fitts' Law (Fitts 1954) models the time $MT$ it takes to move from a particular starting point to a target area as a function of the width $W$ of the target and the distance $D$ to the target:

$$MT = a + b \log_2(\frac{D}{W} + 1)$$

Where $a$ and $b$ are empirically derived constants.

Closely related to Fitts' Law is the steering law (Accot and Zhai 1997), which models the time $T$ taken to navigate a pointing device through a two-dimensional tunnel as a series of Fitts' Law tasks. Represented in general form as:

$$T = a + b \int_C \frac{ds}{W(s)}$$

Where $C$ is the path parameterised by $s$, $W(s)$ is the width of the path at $s$, and $a$ and $b$ are empirically derived constants. Simpler paths may take the generalised form:

$$T = a + b\frac{A}{W}$$

Where $A$ is the length of the path of a constant width $W$. The steering law has been used to accurately predict pointer navigation through corners within tunnels (Pastel 2006) and steering tasks with a variety of devices (Accot and Zhai 1999).

### 2.4.3 Gestures

Interaction can also be made through combinations of multiple keystrokes or mouse interactions—forming gestures. The most common form of gesture is the prolific drag-and-drop, but gestures for specific interfaces (such as the 'flicking' gesture in marking menus (Kurtenbach and Buxton 1994)) and text-entry (such as Unistrokes (Goldberg and Richardson 1993) and ShapeWriter (Kristensson and Zhai 2005)) also exist and have strong spatial properties.

Cao and Zhai (2007) investigated models for user performance in performing gestures and found a high correlation with two models to find the overall time $T$ taken to make a gesture:

$$T = \sum T(line) + \sum T(curve)$$

$$T(line) = aL^b$$

For straight-line segments, where $L$ is the length of the line and $a$ and $b$ are empirically derived constants.

$$T(curve) = \frac{\alpha}{K^{r^{1-\beta}}}$$

For smooth curved segments, where $\alpha$ is the sweep angle of the curve with a radius $r$; $K$ and $\beta$ are empirically derived constants. Cao and Zhai's (2007) work is still a relatively new area of research and does not include allowances for mental perception or processing time, and was not tested against variable error tolerances in interpreting gestures by the system.

## 2.5 PRESENTATION

The presentation of a command interface are the techniques used to expose the previous three classification categories to the user. It is crucial to build an efficient visual interface to support these categories and their factors. There are a number of factors that cannot be modelled by theory: the context, the audience, and the visual style; however, there are factors in the approaches to interface design that can be considered regardless of these subjective factors.

### 2.5.1 Perception

The Gestalt factors and feature integration theory reviewed in section 2.2 apply equally to the visual presentation as they do to the mental model. The perceptual grouping and discernibility afforded by the predictions of these theories allow the user to make efficient pre-attentive analysis and inferences. The use of these theories in the interface presentation can enhance the mental model's cues, strengthening the connection between the user's and the designer's mental model.

Perceptual cues can also take advantage of presentation features such as animation to help guide the relationship between items or ease users between transitions. Bederson and Boltman (1999) found that animation when transitioning between states in an interface improved participants' ability to lean spatial relationships for later recall, without impacting performance.

### 2.5.2 Focus + Context

Furnas (1986) raises the issue of displaying large information structures in a way that allows users to view a great amount of detail about a localised point, whilst preserving information about the global context. Furnas argues for a balance between two aspects: (1) *local detail* for the user to interact with details close to their current focus; and (2) *global context* to tell the user what other parts of the structure exist, and where they are within it.

Furnas's (2006) reflection on these aspects reviews several approaches to "fisheye" view techniques that support such "focus + context" views, exploring degrees of interest and spheres of influence models, arguing that "understanding why [fisheye-degree of importance] subsets might be important for various purposes should help designers know what to consider when turning their designs to varied aspects of users' tasks" (p. 1005). These techniques identify subsets of a large data structure that are important for a given focus—identifying the relevant context.

Closely related to focus + context displays are overview + detail displays. Overview + detail separates the local detail and the global context into distinct displays. This de-coupling allows users to explore the global context without changing the local detail view. These views have typically taken the form of an inset or split window with thumbnails or a reduced-zoom version of the main view (Plaisant *et al.* 1995). Evaluation of overview + detail interfaces has lead to mixed results, but with consistently strong user satisfaction (Hornbæk and Frøkjær 2003, Hornbæk *et al.* 2002).

### 2.5.3 Customisation

Findlater and McGrenere (2004) discuss the difference between *adaptable* interfaces that can be customised by the user, and *adaptive* interfaces that automatically customise themselves in response to the user. In empirical studies, they found that participants preferred an adaptable menu over adaptive or static menus, but commented that effective customisation was guided by example (in the case of participants who used an adaptive menu interface first) rather than easy-to-use controls. Gajos *et al.* (2006) further explored adaptive interfaces and found that participants preferred passive adaptations that did not spatially disrupt the rest of the interface. Page *et al.* (1996) presents an analysis of how users customised a word processor's interface and gave several recommendations for designers—(1) expect users to customise; (2) make customisation easy; (3) make customisation fit the work, maintaining user's workflow and "rhythm"; and (4) optimise defaults for casual users.

## 2.6 SUMMARY AND DESIGN CONSIDERATIONS

From the research presented in the preceding sections, we can draw design considerations that are applicable to command interfaces. These considerations follow the classification hierarchy outlined in section 2.1 and figure 2.1. The emphasis with this classification is to give designers a set of considerations that interfaces can be evaluated against.

### 2.6.1 Organisation

**Perception** The user's ability to perceive and can infer relationships based on the organisational structure of a command interface.

- **Grouping** The grouping of objects through various explicit and implicit Gestalt factors. Organisational cues can assist users to pre-attentively identify groups of objects as a related set of commands and infer relationships between commands.

**Feature Distinction** The distinction and similarities between objects in their perceptual primitives. Both extremely low levels and extremely high levels of distinction makes identifying specific objects difficult, but a balance that places appropriate emphasis across items facilitates meaningful discernment between items and groups.

**Consistency** Maintaining consistency in the organisation of related sets of objects helps in the perceptual grouping of commands; similarly, keeping a level of discontinuity between groups helps emphasise their distinction. Conversely, continuity can assist in the formation of perceived relationships between commands.

**Mental Modeling** The ability for users to form a mental model of the command structure and for that mental model to hold and support interaction with the structure.

**Mapping** The strength of the relationship between the presented physical model and the user's perceptual model of a command structure. Strengthening this relationship reduces the mental workload for users to locate information within the structure as navigation becomes more 'natural'.

**Visual Momentum** The ability for users to infer connections and relationships when moving around the structure, based on certain features.

**Landmarking** Assisting visual identifiers that anchor the user around a point or provide an assisting transitional identifier when moving between two related interfaces. Landmarking hierarchies is context-dependent as to the representation, location, and quantity of them—but effective landmarking allows for the rapid identification of central features and continuity when transitioning across the structure.

**Integration** The ability for users to mentally integrate subsets of commands into the overall command structure—for example, integrating commands accessible by both menus and toolbars into the overall command hierarchy. Transitioning between various disparate representations of command interfaces that are part of the same command structure is a mental task that can be eased if the command structure makes the mental integration between multiple representations more obvious and continuous.

**Connectivity** The organisational structure can be modeled as a directed graph; there are various properties of this graph that maximise the ability for users to locate and discover items. There is a large degree of context-dependence in determining how much connectivity to give to a command structure. Too much can overload the user with options, whilst too little can induce serial hunting tasks to find items.

**Redundancy** The amount of redundancy in the structure to maximise node reuse. Increasing the level of redundancy can assist with the formation of a mental model by identifying distinct groups of commands, and reduce navigation time to frequently redundant commands.

**Links** The number of connections between nodes, maximising the possible number of paths. Connections between nodes in the graph assist with navigation, but can also act as an assisting tool in building a spatial representation of the command structure.

### 2.6.2 Navigation

**Traversability** The ability for a user to move around the structure being navigated. Navigation systems should allow for as much freedom of movement as possible when the user is traversing the system, but wary not to hinder navigation and searching tasks.

**Locomotion** The speed and flexibility for moving around a structure, and the degree of control the user has over this. Navigation tools need to allow for a change in navigation style as users switch between searching and recall tasks.

**Mutability** The ability for users to manipulate their navigation view when moving around it, or the mutation of the view by the system for the user's benefit. Users should be free to customise or choose the tools they use for navigation in their particular workflow.

**Navigability** The ability for a user to find a particular item in a structure. In contrast to traversability, the focus with navigability are the aids for hunting tasks when the user is trying to locate a specific item or command in the structure.

**Wayfinding** The ability for the user to move through the items in a structure to find a particular target. This combines information about what the user knows about the structure (their mental model) and the information being given to them by the system about where they are and where they can navigate to.

**Information Scent** The strength of cues in the structure to indicate the location of a navigational path to another item in the structure. At the very least, items should have scent about where their immediate connecting paths lead, and at best about every other item accessible from the current item—a balance has to be found between strong navigability and relevance of the cues.

**History** The ability for the user to return to locations they had previously visited in the structure. This differs from 'back' and 'forward' mechanisms that work on the current navigation context by allowing the user to move to arbitrary locations that they had previously visited (in the current navigation session or otherwise).

### 2.6.3 Interaction

**Pointing** The difficulty and time taken for the user to acquire a target.

**Size** The on-screen dimensions of the target. Larger targets are easier to acquire, as are targets that utilise the properties of the visual display—such as the screen edges, which have effectively infinite size.

Distance The distance of the target from the current cursor location. As described by Fitts' law, targets that are large and close are fastest to acquire.

Path The path that the user must navigate through to reach a target. The difficulty of pointing to a target increases with the complexity of the path that needs to be taken to acquire a target (as modeled by the steering law).

Decision The time to select a target, as modeled by the Hick-Hyman law. The fewer probable targets there are for the user to decide between, the faster the user can move to the acquisition of the target.

Probability The probability of a target being selected. This is the probability that a particular command fulfills a user's task.

Discernibility Although not modelled by the Hick-Hyman law, users must be able to distinguish between alternatives. This is predominantly fulfilled by presentation techniques described in section 2.6.4.

Gestures The complexity and shape of the gestures required to issue a command. Simpler gesture paths are easier and faster to perform than long, complex ones; however, spatially relevant gestures can reinforce their spatial learning.

### 2.6.4 Presentation

Perception The perceptual effects of the presented interface, as presented for organisational models in section 2.6.1.

Focus+Context Focus + context in command systems pertains to the ability of the user to orientate themselves in the command system, and effectively have the tools to perform a specific task whilst having the context to easily move between command sets as they move between tasks.

Local Detail The amount of local detail the user has readily accessible and can interact with. In a command interface, there should be enough local detail (commands) to allow the user to perform as much of a given task as is effectively possible.

Global Context The level and detail of global context the user has to orientate themselves and explore. Global context should make it easy to identify the local detail a user is currently viewing, the hierarchy surrounding it, and allow identification of other relevant command sets.

Customisation How the interface can be customised to meet the needs of the user's workflow. Two forms of customisation are available, for which a balance should be found between.

Adaptable The amount of user-driven customisation of the interface that can be performed. Adaptable interfaces must not only be easy to customise, but the value of customisation should be made apparent to the user—for example, through example or suggestion.

**Adaptive** The amount of system-driven customisation that is performed in response to user behaviour. Any adaptive interfaces should be passive in their appearance and operation—keeping spatial stability in the rest of the interface to avoid disorienting the user and corrupting their spatial knowledge of the interface.

CHAPTER 3

# Principles of Navigation

The research presented in chapter 2 gives us a foundation from which we can build new theories applicable to command interfaces. This chapter will focus on one aspect of the classification outlined in section 2.1—navigation. 'Navigation' defines the properties of how users traverse through an interface ("wayfinding + locomotion" (Jul and Furnas 1997)), and is influenced by the organisational structure to be navigated, the presentation of the structure, and the tools used to traverse it. We present the principles of interface navigation that influence user efficiency, building upon the design considerations, presented in section 2.6.2, of organisation, interaction, and presentation issues.

We describe interfaces for navigation as constructed from three core components: (1) *a structural graph* that identifies the items to be navigated and describes the relationships and connections between them; (2) *views* that visualise a structural graph to the user; and (3) *tools* that allow the user to manipulate and traverse views. The following sections describe these components and the factors that influence a user's performance with them.

## 3.1 EFFECTIVE VIEW NAVIGABILITY

Our principles use Furnas's (1997a, 1997b) model of navigation—effective view navigability (EVN)—reviewed in section 2.3.2, as a starting framework for describing navigation influences and focus on describing user interaction in four areas: (1) the structure of what is being navigated (organisation); (2) the views of a particular structure (presentation); (3) the tools used to manipulate the views (navigation, interaction); and (4) the interaction between these areas.

EVN supports modelling many simple navigation tasks but starts to show shortcomings when applied to more complex navigation interfaces. EVN's primary focus is on modelling the structure as a 'viewing graph' and subsequent navigation graph, but does not give enough consideration to the presentation of such graphs (the 'views' that visualise a graph to the user). EVN assumes a single view for each graph, which is

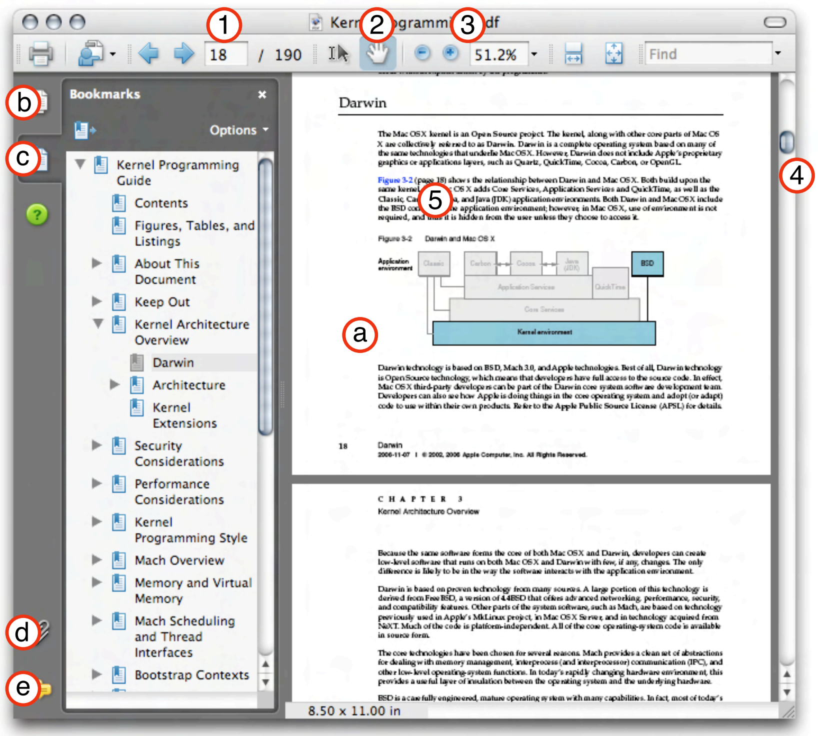



*Figure 3.1*: A document open in Adobe Reader 8. There are five labelled navigation views of the document available: (a) the document view, (b) page thumbnails, (c) document bookmarks, (d) attachments, and (e) annotations. Additionally, there are five labelled navigation tools: (1) page navigation, (2) pan tool, (3) zoom control, (4) scrolling, and (5) hyperlinks.

appropriate for zooming or focus + context interfaces, but does not consider scenarios with multiple views of the same graph, such as those in overview + detail interfaces where there may be interactions between the views. For example, document viewers often have several displays of the document content sampled at various levels—the interface shown in figure 3.1 has several available views of the document. Each view may be synchronised in some way with another and each may manipulate other views, the impact of which is not discussed in EVN.

There is also little consideration given to the tools used to manipulate views of navigation graphs. Navigation tools are the primary device for manipulating and traversing views and thus have strong implications for the navigability of an interface.

## 3.2 THE STRUCTURAL GRAPH

From a designer's perspective, every navigable structure starts with an underlying graph representing the structure and connections between each item. The structural graph defines the position of each element in relation to its parents, siblings, and children. For example, graphs for lists and documents would typically be a serial line graph connecting each item to the one immediately preceding and one immediately following it, such as that visualised by the main document view in figure 3.1; command structures would exhibit a more hierarchical structure to group and classify commands, such as that shown in figure 3.4. It is important to emphasise that the structural graph is a not a visual representation, it is the abstract organisation of what will be interpreted by navigation views. The structural graph is strongly influenced by the design issues raised in section 2.6.1, and parallels EVN's logical structure and viewing graphs.

This graph is the basis for the mental model that the user will develop as they use the interface. As reviewed in section 2.2.2, designers present this structural graph through the user interface, rather than communicating it directly to the user. The user's ability to accurately form the mental model intended by the designer is key in aiding their understanding of the relationships and ultimately, their interaction with the model. The visual user interface is only one aspect in helping the user to form an accurate mental model, the structure of the model itself has to be understandable and 'natural' to the user.

### 3.2.1 Connectivity and Complexity

There will often be more than one representation of the structural graph due to the multiple ways of describing connections between items. We have identified three different types of connections that may appear in a structural graph (illustrated in figure 3.2):

1. *Logical Connections* The logical connections are the 'natural' connections between items. For example, the parent/child relationship between a group and its contained items can be represented as a hierarchical tree, the edges denote the relationship between the items. This can commonly be seen in cascading menu hierarchies or in linear documents that contain hierarchical structures (such as chapters, sections, and subsections).

2. *Shortcut Connections* Connections can be made between arbitrary locations in the graph (or to other graphs) that do not follow a natural relationship between each item. Hyperlinks are an example of such links that allow users to 'jump' between documents or sections of documents that don't share a logical connection. These shortcut connections are often unidirectional with no facility in the graph to follow a link backwards (this is commonly a feature of the navigation tools).

3. *Composite Connections* One area of the graph can be referenced by another part, causing the referenced items to be 'included' as part of the graph at the point the reference is made. The items are not actually at that point, but navigation through the composite connection makes it appear so. For example, the HTML `<iframe>` element[1] allows designers to embed portions of a document within another.

   Each use of these connections increases both the connectivity to, and redundancy in, areas of the graph, but at the cost of theoretical complexity when traversing the graph due to the larger number of connections that must be considered. EVN makes two recommendations to combat this complexity that are applicable to structural graphs (Furnas 1997a):

> **Small views** "The number of out-going links, or out-degree, of nodes in

[1] `http://www.w3.org/TR/html401/present/frames.html#h-16.5`

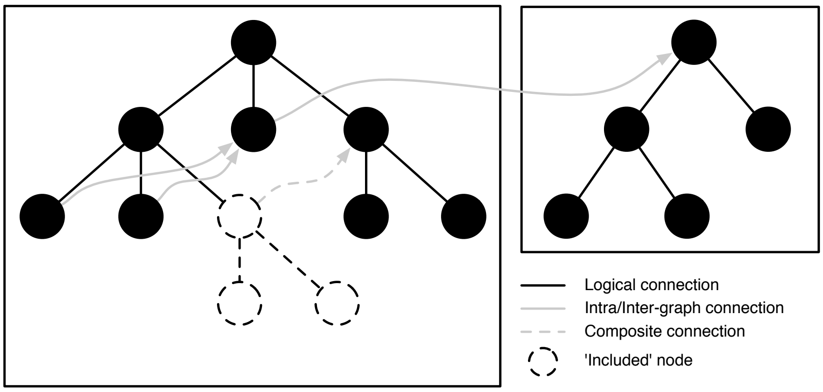



*Figure 3.2*: An example of the four different types of connection between nodes.

the [graph] must be 'small' compared to the size of the structure." (p. 368)

**Short paths** "The distance [in number of connections/edges] between pairs of nodes in the [graph] must be 'small' compared to the size of the structure." (p. 368)

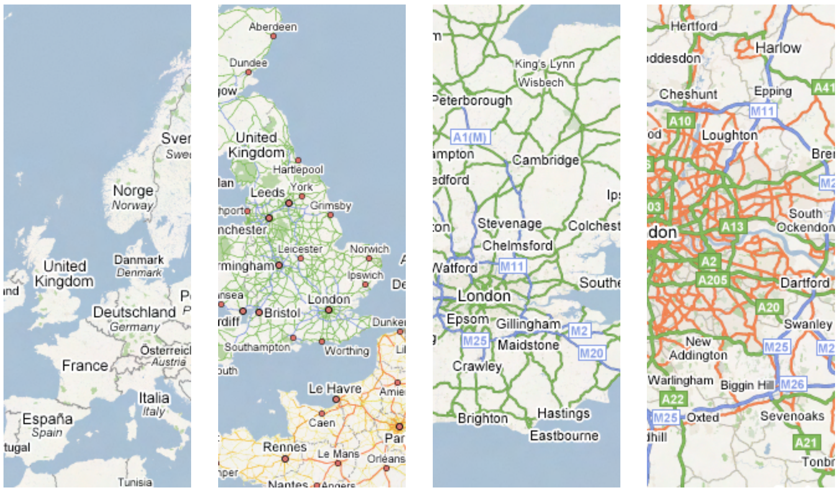
Connections that are not 'logical connections' can violate the natural structure that users form their mental model around—increasing the mental workload for the user in building an understanding of the relationships between points in the graph. Shortcut and composite connections are useful navigational aids if they *support* the logical structure rather than define it. For example, shortcut connections are often unidirectional, with no facility in the graph to support navigation back to where the user navigated from (there is commonly a reliance on navigation tools to provide this). In contrast, bi-directional shortcuts create connections that avoid 'stranding' the user at an unfamiliar point in the graph by giving the user a connection back to familiar territory. We want to use shortcut and composite connections to promote connectivity in the graph, but not at the expense of the user's mental model. From this, we propose an additional principle:

**Minimise complexity** Connections should not define or violate the logical structure of the graph. There should always be a facility for the user to remain connected to familiar territory.

We want to ensure that at all times, the user knows where they are, where they came from, and where they can proceed to.

### 3.2.2 Landmarking

One of the issues raised in section 2.6.1 is the use of landmarks—visual identifiers or cues that assist the user in locating and orienting themselves around a point in the graph. Landmarks can be existing points in the graph, or shortcuts to particular points but not




(a) Major country names.

(b) Country names and major cities.

(c) Cities, districts, and major motorways.

(d) Motorways, main roads, cities, and major towns.

*Figure 3.3*: Google Maps, demonstrating dynamic landmarks as the zoom of the view changes, but that remain static for that zoom level. Continuity between zoom levels is maintained by providing landmarks that exist at multiple zoom levels.

actually part of the graph. Techniques for successful use of landmarks for navigation are suggested by Woods's (1984) visual momentum theory, reviewed in section 2.2.2.

Landmarks should be spatially stable points that a user can rely on being constant; however, they also need to be context-dependent on the current focus of the graph. For example, Google Maps[2], shown in figure 3.3, changes which landmarks are visible depending on the level of zoom (from major country names in (a) to motorways and major towns in (d)); but the number and location of landmarks at a particular zoom level does not change as the user pans around the map—landmarks are spatially fixed to a particular point on the map. Continuity between each zoom level is maintained by supporting the visual momentum technique of 'perceptual landmarks' and the use of semantic zooming (Perlin and Fox 1993), enabling a smooth perceptual transition between levels whilst activating new landmarks relevant to the new zoom level. To this extent, we propose the following recommendation:

> **Visual momentum** When transitioning between foci, landmarks should utilise visual momentum and semantic zooming techniques.

Effective landmarking is vital in assisting the user visually search large structures (Hornof 2001), but presenting the user with too many landmarks can decrease their usefulness as the increased clutter induces additional navigation and visual search tasks. The user should be able to perform a parallel, pre-attentive scan of landmarks (as de-

[2] `http://maps.google.com/`

scribed by feature integration theory—reviewed in section 2.2.1); the use of landmarks should not induce serial hunting tasks. To keep landmarks as useful as possible, even as the size of the graph increases, we propose a recommendation:

> **Landmarks** There should be a logarithmic relationship between the number of visible items and the number of displayed landmarks.

This promotes landmarks which are general enough to describe a large set of items, whilst keeping the set of landmark items low enough to remain useful navigation aids for the user. This relationship is evaluated in chapter 4.

## 3.3 NAVIGATION VIEWS

Navigation views are the devices used to present a structural graph to the user, and can vary greatly in their presentation—from document viewers (figure 3.1) to menus and toolbars (figure 3.4). Navigation views will typically display an interpretation of a graph, rather than a literal display; for example, figure 3.1 shows Adobe Reader[3] with several navigation views of a single structural graph (the open document)—view (a) shows a serial graphical representation of each page, whereas view (c) has sampled the graph for chapter and section headings.

As described in section 2.5.2, there are three forms that these views take: (1) focus + context, (2) overview + detail, and (3) zooming/panning. Each of these views strive for a balance between global context and local detail (reviewed in section 2.5.2) to enable the user to view and find paths to the information they want.

### 3.3.1 Local Detail and Global Context

The balance between local detail and global context is crucial in presenting a navigable view. Local detail needs to be at a resolution that is usable and plentiful enough to be useful to the user; global context needs to give enough information to allow the user to purposefully navigate to other parts of the graph.

The display of local detail and global context can take many different forms—for example, Adobe Reader (figure 3.1) utilises one view dedicated to local detail and several dedicated to global context with bi-directional synchronisation between them. Hierarchical menus (figure 3.4) use a single view to show both the local detail of the current menu and a 'trail' of global context to the top-level menu item; global context about other menus varies depending on which menus are open (utilising a 'spheres of influence' model—see section 2.5.2).

### 3.3.2 Wayfinding

Navigation views are instrumental in assisting the user in discovering paths between their current location and where they want to move their focus to. This is done through the effective use of information scent (reviewed in section 2.3.3), where accurate hints about the contents of remote parts of the structure is given. A simple example of this is shown in figure 3.4 where menus with cascades provide scent about the cascade's

[3] `http://adobe.com/products/reader/`

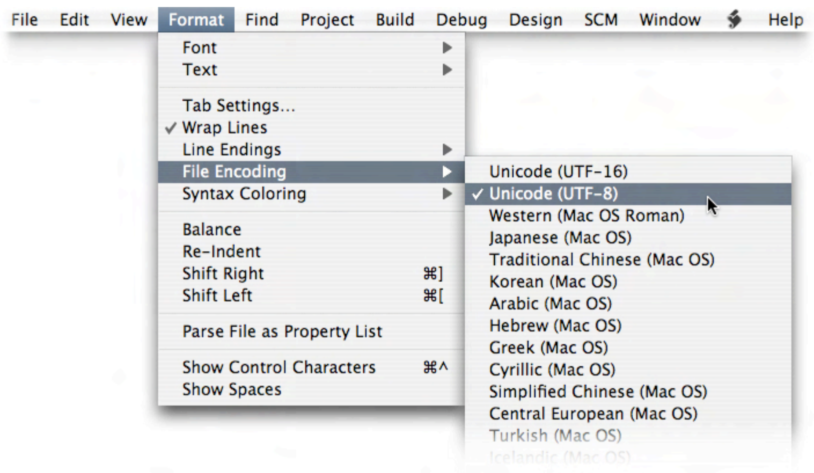



*Figure 3.4*: A hierarchical menu—a 'panning' style navigation view. Local detail is shown through the currently open menu, with global context in the 'trail' of open menus back to the starting point; but no context is given about menus until they are opened, only a scent in the description.

existence through the right-facing triangle and the content of the menu through their titles.

In order for a view to be navigable, we propose the following recommendation:

> **Wayfinding scent** A navigation view must give scent about all items not in focus. The scent must provide enough information to both identify *what* the item is, and *where* it is located.

A view that contains scent about every item not in focus would be completely navigable, as a user would be able to accurately locate any item from any location. However, there are practical considerations about the resolution of such information scent—as with landmarks and local detail/global context, a balance must be found.

### 3.3.3 Multiple Views

Multiple views of a single structural graph are a characteristic of overview + detail and often employed in zooming/panning interfaces to visually separate local detail from global context (Cockburn *et al.* n.d.). For example, Apple Pages[4] can display a strip of page thumbnails alongside the main document view, as detailed in figure 3.5. Each of these views are independent visualisations of the graph, but each can interact with and manipulate each other—a user's actions in one can be reflected in another. Synchronisation between these views is typically such that more general (overview) views are independently explorable without immediately influencing more detailed views; in

[4]`http://www.apple.com/iwork/pages/`

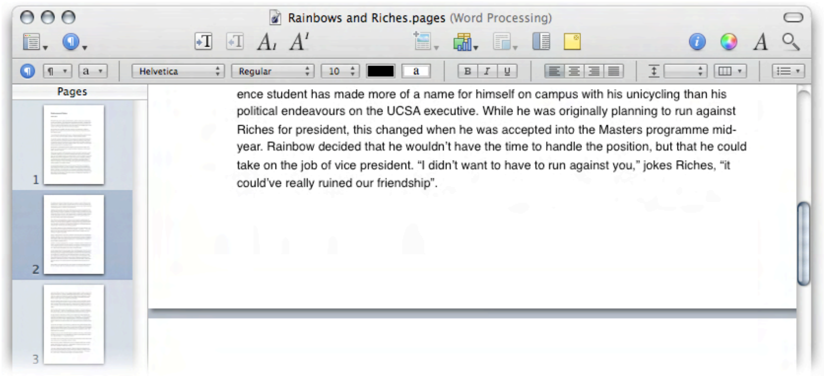



*Figure 3.5*: Apple Pages 3's overview + detail display: A view of page thumbnails is available alongside the main document view. The highlighted page in the thumbnail view is synchronised with the page shown in the main document view, but the thumbnail view can be navigated independently of the main document view.

contrast, changes to detail views are often immediately reflected in more general views. However, the synchronisation between multiple views is dependent on the user's expectations of interacting with a particular structural graph.

Following from the observations of Cockburn *et al.* (n.d.), we present the following recommendation:

> **Multiple views** The relationship between multiple views of the same structural graph should be such that synchronisation between views is maintained, but more general views should allow for independent user exploration.

## 3.4 NAVIGATION TOOLS

Navigation tools are the utilities that manipulate a navigation view. Navigation tools share a common goal with navigation views—they both need to provide a way for the user to successfully find the information they want. However, they differ in that tools are not about the display of information, but about controlling the presentation. There are three types of navigation tools: (1) tools that manipulate the focus of the view (such as scrollbars and pan tools); (2) tools that manipulate the the scope of the view (such as zoom controls); and (3) tools that manipulate the sampling of the view (such as disclosure triangles). All of these tools aim to provide the best balance between speed of movement and accuracy of location in allowing the user to traverse across a structural graph.

### 3.4.1 Speed and Accuracy

The design of navigation tools must find the balance between allowing users to navigate quickly and allowing them to navigate accurately. Scrollbars are an example of such navigation tools where this balance has been explored at length by research (Wallace

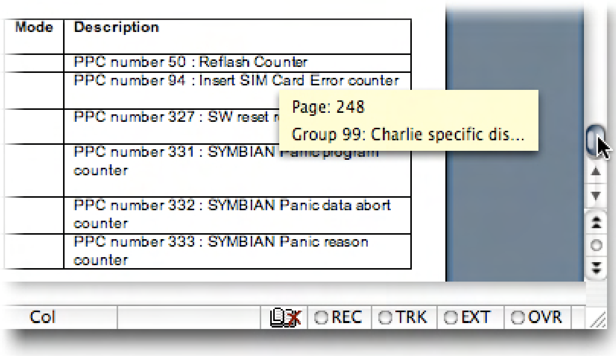



*Figure 3.6*: Scrolling in Microsoft Word 2004. A small tool-tip appears to provide context about the current location of the scroll widget without the user having to pause scrolling to check the contents of the document.

*et al.* 2004, Cockburn *et al.* 2005, Cockburn *et al.* 2006).

Traditional scrollbars provide a very fast mechanism to navigate between points in a document but are not very accurate due to the limited feedback given by the scroll widget of the user's current location (only positional feedback, not contextual). Microsoft Word[5] (shown in figure 3.6) provides a tool-tip when scrolling that indicates the page number and section title of the current page that the scroll widget is positioned at; improving the accuracy of scrolling at a small cost of speed due to the increased demand on attention and perception to utilise the information provided by the tool-tip.

The mechanism typically used for improving accuracy is to increase the amount of information scent given to the user about the outcome of their actions. However, information scent can erode user performance if it is inaccurate, unnecessary, or distracts them from their task. As users make a transition from novice to expert, the information scent becomes less important to successfully navigate rapidly. Together with the guidelines already presented for information scent, we give an additional one for the scent used in navigation tools:

**Tool scent** Information scent used in navigation tools must be passive and promote, rather than obstruct, a user's transition from novice to optimal expert behaviour.

## 3.5 CONCLUSIONS

The preceding sections have presented a set of principles for navigation interfaces that build on the research presented in chapter 2 and Furnas's (1997a, 1997b) EVN model of navigation. The three core components of navigation interfaces (the structural graph, navigation views, and navigation tools) each have their own principles that guide their influences over an interface's navigability. Each component also interacts with each other and with the user. It is necessary to understand these components and their in-

[5] `http://www.microsoft.com/mac/products/word2004/word2004.aspx`

teractions to design an efficient user interface; but it is also important to stress that there is a degree of context-dependence in the *application* of the presented principles. We believe that these principles can be used to form models of interface navigation and will be explored in such a way in the following chapters.

CHAPTER 4

# Evaluation

Chapter 3 presented a set of principles that we believe influence user performance with interfaces for navigation. Formal empirical evaluation is required in order to establish the validity of these principles. Several of the presented principles have already been supported by existing evaluation, which has been reviewed in chapters 2 and 3. However, the issue of landmarking has not be thoroughly studied by prior research. We have conducted a quantitative analysis of the *landmarks* principle given in section 3.2.2.

The following sections detail the evaluation of our principle for landmarking structural graphs; our proposed technique is evaluated against several other strategies for landmark visibility. We found that our proposed technique was significantly better than several other strategies for landmarking.

## 4.1 EVALUATION OF LANDMARKS

Section 3.2.2 presents a guideline for displaying landmarks to improve the navigability of a structural graph that states:

> **Landmarks** There should be a logarithmic relationship between the number of visible items and the number of displayed landmarks.

This guideline has implications for a wide range of navigation interfaces where landmarks are used to aid navigability.

Interfaces that necessitate the use of landmarks for navigation due to the unpredictable, yet clustered, nature of the data being presented are the targets of this guideline. For example, maps or topological visualisations often present data that is organised into groups, but for which there is only a neighbourly relationship between each group. For these interfaces, the use of landmarks is essential in assisting users locate items and orient themselves within the data set.

As discussed in section 3.2.2, landmarks serve as navigational aids and should not obstruct a user's navigation by inducing additional visual search tasks. We believe that these goals can be best supported through a model that describes the ideal number of landmarks to be presented, in terms of the current number of visible items from the structural graph.

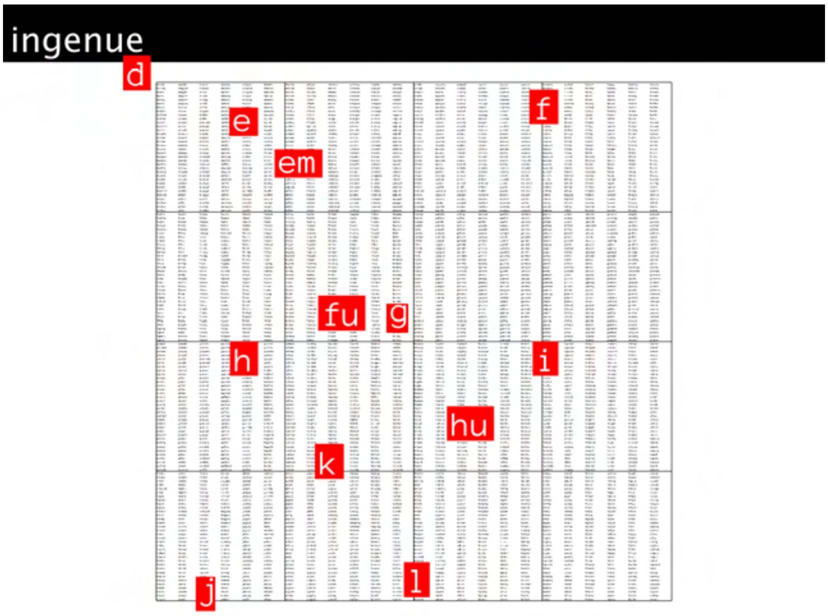



*Figure 4.1*: The evaluation interface prompting the user to select the word "ingenue" in the logarithmic condition.

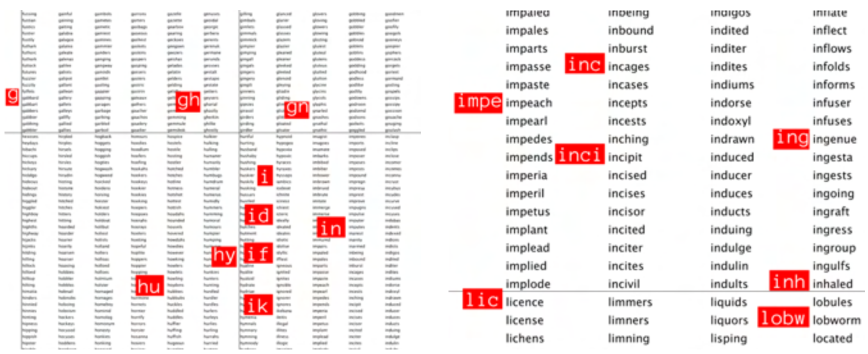
We aim to show that the use of a logarithmic relationship between the number of visible items and the number of visible landmarks best facilitates navigation by allowing pre-attentive searching of landmarks. Our evaluation will compare four functions $f(n)$ that describe the ideal number of landmarks to be shown for $n$ visible items: (1) *square-root*: $f(n) = \sqrt{n}$—chosen due to the high number of landmarks that will result; (2) *logarithmic*: $f(n) = \log_2 n$—chosen for the smooth incline produced; (3) *seven*: $f(n) = 7$—loosely based on Miller's (1956) research of short-term memory's capacity for storing "chunks" of information; and (4) *zero*: $f(n) = 0$.

Tasks in the experiment involved locating and selecting a word from a grid of 3,264 alphabetically arranged words. Landmarks were used to denote the location of word ranges within the grid.

### 4.1.1 Interface

The evaluation environment emulated an interface where participants had to zoom and pan to locate a target. As the level of zoom changed, the landmarks changed to reflect the current set of items shown to the user. A screenshot of the evaluation interface is shown in figure 4.1.

Words were shown alphabetically in a $4 \times 4$ grid of cells (with the exception of training conditions, shown in a $2 \times 2$ grid). Each cell consisted of 6 columns of 34 words each; each word was rendered in a $180 \times 32$ unit area. Cells were arranged from left-to-right, top-to-bottom in the grid with a solid black dividing border.




(a) (b)

*Figure 4.2*: Landmarks as the user zooms between (a) and (b).

Zooming was controlled by moving the mouse wheel away from the user (to zoom in), or towards the user (zoom out). Each zoom action was performed around the position of the mouse cursor (such that the point under the cursor remained under the cursor after the zoom adjustment) and altered the current width and height of the viewport by a factor of $\sqrt{2}$—chosen after examining the zooming properties of several other document navigation interfaces. At any zoom level, participants could pan by holding down the left mouse button and dragging. There was a 1:1 mapping ratio between cursor motion and pan distance.

Selection was performed with a single left-click within a word's area, and could be performed at any zoom level.

Words were rendered in black-on-white Lucida Grande; landmarks were rendered as white-on-red blocks of monospace Anonymous[1]. A monospace font for landmarks was used to ensure a consistent size between landmarks of identical length.

### *Landmarks*

Landmarks were used to denote a range of words, rather than a specific word in the set—for example, words beginning with '*b*', or words beginning with '*bea*'. Landmarks were chosen based on the content of visible items—at any particular zoom level, a list of possible landmarks was generated (all landmarks for word ranges that began in the list of visible words); landmarks were then selected from the list in the following order:

1. Landmarks that had previously been rendered at the current or more distant zoom levels.
2. The possible landmarks were then filtered by length into groups. Landmarks were then randomly selected from each group (in ascending order); when a group was exhausted, selections continued randomly from the next group (in ascending order).

This continued until the maximum number of landmarks prescribed by the current condition had been selected (shown in figure 4.2).

[1] `http://www.ms-studio.com/FontSales/anonymous.html`

This ensured that landmarks were always as general as possible in terms of their descriptive power; and that transitions between zoom levels were as smooth as possible by employing the visual momentum technique of perceptual landmarks.

Landmarks were always displayed at a fixed size, regardless of zoom; and were always displayed to the left of the first item (from the entire set of words) of the word range indicated by the landmark.

### 4.1.2 Participants and Apparatus

The fourteen volunteer participants (twelve male, two female) were all students from the University of Canterbury. The experiment took approximately twenty minutes to complete for each participant.

The experiment ran on an Intel Pentium D 3 GHz computer running Fedora Core 6, equipped with 1GB of RAM and an NVIDIA GeForce 6200 connected to a 17″ LCD display at $1280 \times 1024$ resolution. All input was through a Microsoft Wheel Mouse Optical, with a 1:2 control-display gain ratio.

Python/OpenGL software controlled the participants' exposure to conditions; the software ran full-screen, and logged all actions to microsecond granularity.

### 4.1.3 Experimental Design and Procedure

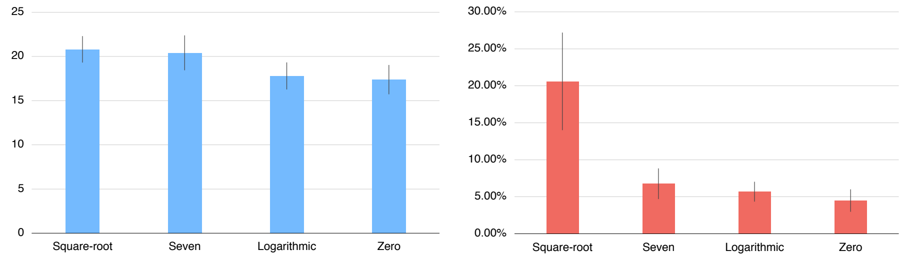
#### *Interface Familiarisation*

A demonstration (approximately two minutes) of the interface was given to participants before they began the experiment. The demonstration described the cueing environment, how to manipulate the interface, and the nature of the landmarks shown; participants were encouraged to ask questions if they were unsure of the interface controls before starting. During the experiment, all participants were given a training session of three selection tasks (identical in nature to those in the timed evaluation, but using a smaller, different set of words) prior to each condition to ensure they were familiar with the controls and the density of landmarks that would be present in the timed selections.

#### *Procedure*

Each condition consisted of seventeen random selection tasks and conditions were counter-balanced with an incomplete Latin square. Each selection task began at an initial zoom level such that every item was visible on-screen (as shown in figure 4.1). The user then had to zoom/pan until they could locate the target item. Upon a correct selection, the zoom was reset to the initial level and the next selection was prompted. An incorrect selection caused the background of the selected word's area to momentarily turn red, but timing continued until the correct selection was made.

All tasks were completed with one condition before beginning with the next, with a voluntary rest period between each condition. At the conclusion of the experiment, participants were asked to complete a subjective evaluation form.




(a) Mean selection time (seconds) per condition (+/- SE)

(b) Mean error rate per condition (+/- SE)

*Figure 4.3*: Target selection times and error rates.

### *Words*

Words were selected from a list of 18,553 seven-letter words[2]. The word list was alphabetically divided into four, 4,638 word blocks from which 3,264 words were randomly selected for each condition. This division was to ensure the elimination of learning effects as to the spatial locations of word ranges across conditions. Training conditions had 816 words randomly selected from a list of five-letter words[3].

### *Design*

The primary dependant variable is task completion time: the elapsed time from revealing the stimulus word until its correct selection. The experiment was run as an analysis of variance (ANOVA) for the within-subjects factor *landmark function* with four levels: zero, seven, logarithmic, and square-root.

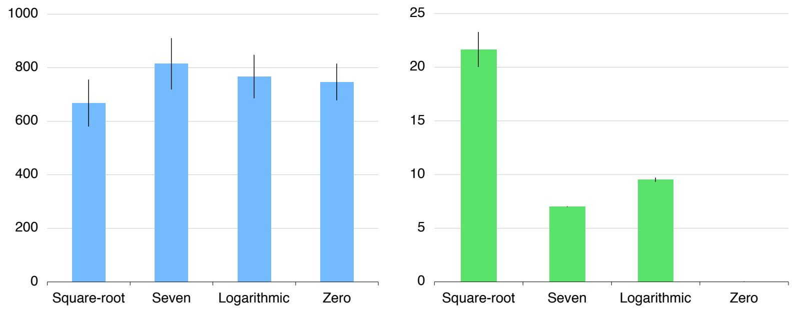
## 4.2 RESULTS

### 4.2.1 Empirical Results

A summary of the empirical results are shown in figure 4.3. There is a significant main effect for the factor *landmark function* ($F_{3,39} = 9.862$, $p < 0.001$). The zero landmark condition allowed for the fastest mean selection time of 17.37 seconds (sd. 6.21), followed by logarithmic (17.79s, sd. 5.72), constant (20.41s, sd. 7.41), and square-root (20.80s, sd. 5.58). A post-hoc Tukey test gives an Honest Significant Difference (HSD) of 2.47s ($\alpha = 0.05$). This reveals a number of pair-wise significant differences; notably, a significant difference between the logarithmic and "seven" conditions.

Error rates were highest in the square-root condition with a mean error rate of 20.59% (sd. 24.71%), followed by seven (6.75%, sd. 7.75%), logarithmic (5.67%, sd. 5.01%), and zero (4.48%, sd. 5.67%).

[2] Retrieved from `http://www.math.toronto.edu/~jjchew/scrabble/lists/common-7.html`
[3] Retrieved from `http://www.math.toronto.edu/~jjchew/scrabble/lists/common-5.html`




(a) Mean number of visible items per condition (+/- SE)

(b) Mean number of landmarks per condition (+/- SE)

*Figure 4.4*: Average number of items/landmarks visible to each participant during each condition.

### 4.2.2 Subjective Results

Participants were asked to rank the interfaces in order of preference for the particular landmarking style (ranks 1 to 4, best to worst); interfaces were described to them as "no landmarks" (zero), "few landmarks" (seven), "some landmarks" (logarithmic), and "many landmarks" (square-root)—with a small screenshot representing each one. The logarithmic condition had the best ranking with a median of 1, followed by constant, zero, and square-root (with medians of 2, 3, and 4 respectively). The poor ranking of the square-root condition was backed by negative comments about the clutter and visibility issues caused by the large number of landmarks.

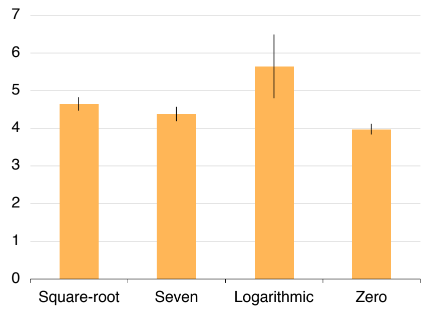
Other comments by participants noted the lack of useful landmarks in the "seven" condition and the need to perform two visual searches in the square-root condition—one to search for a landmark, and another to search for the target.

### 4.2.3 Navigation Data

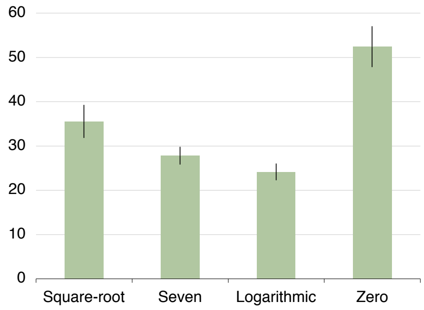
In addition to the selection timing, data was also logged on participants' navigation actions during the timed selections. We can analyse this data to look for differing navigation behaviour characteristics between each condition and correlate this with our empirical results.

#### *Visibility of Landmarks and Items*

Figure 4.4 summarises the average number of items and landmarks shown to users in each condition. Data points were gathered after each zooming or panning action. Higher numbers of visible items indicate that users were consistently at a further level of zoom when they were navigating; no significant trends were found. For landmarks, the square-root condition displayed the most landmarks to the user, with a mean of 21.65 (sd. 6.10) landmarks visible after each zoom or pan action; followed by logarithmic, with a mean of 9.50 (sd. 0.74) landmarks.

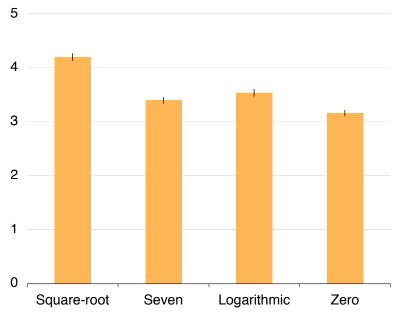



(a) Mean number of zoom actions each selection per condition (+/- SE)

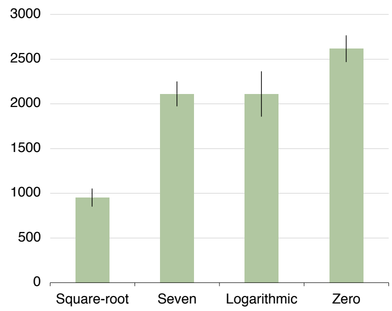



(b) Mean number of pan actions each selection per condition (+/- SE)

*Figure 4.5*: Average number of zoom/pan actions made by each participant during each condition.




(a) Mean zoom level when participants made their correct selection per condition (+/- SE)




(b) Mean pan distance for each selection per condition (+/- SE)

*Figure 4.6*: Average zoom and pan distance for each selection per condition

### *Panning and Zooming Actions*

Figure 4.5 summarises the average number of actions used by participants in each condition. Participants performed significantly more panning operations in the zero condition (mean of 52.42, sd. 71.20) than any other condition. However, the high standard deviation indicates a wide degree of variance in the behaviour between participants—a similar trend exists for zooming actions.

The actual zoom levels and distances panned by participants is summarised in figure 4.6. The zoom level is measured as the number of "zoom in" actions it would take to achieve the level of zoom participants did when they made each correct selection. Selections in the square-root condition were performed at a significantly closer zoom level (mean of 4.19, sd. 1.10) than any other condition; in contrast, participants panned the least in the square-root condition (mean of 950.45 units, sd. 1567.68) and the most in the zero condition (mean of 2615.57 units, sd. 2314.28). As with zooming/panning actions, the high standard deviations show a variance between participants.

## 4.3 DISCUSSION

The results shown in figure 4.3 support our hypothesis for a logarithmic relationship to determine landmark visibility—with the logarithmic condition performing significantly better than both square-root and constant ("seven") landmarking conditions. The logarithmic condition was further supported by the subjective results.

We believe that the strong performance observed for the zero landmark condition was due to a combination of the predictable organisation of the items and the lack of distractors in the condition. This is not an issue for our hypothesis, as the target application of our principle are interfaces that necessitate the use of landmarks due to the unpredictable nature of their organisation. The lack of significant difference between the logarithmic and zero landmark conditions show that the use of logarithmic landmarks did not induce additional visual search tasks upon the user. This will be further discussed in chapter 5.

From figure 4.4 we can see that, on average, the logarithmic condition resulted in more visible landmarks than both the square-root and seven condition, yet the user performance does not carry the same characteristic. This reveals that performance was not dictated solely by the number of landmarks visible, but that it was a contributing factor. In the case of the "seven" condition, the sparse landmarks were often unhelpful and acted as distractors to the task. In the square-root condition, the abundance of landmarks induced their own visual search tasks that needed to be completed before searching for the target item.

We believe that the logarithmic condition was able to find the appropriate level landmarking that allowed for useful landmarks (further supported by the comments from several participants)—preventing them from being distractors to the task. However, further study needs to be conducted to assess why the factors that caused the "seven" condition to perform significantly worse than the logarithmic condition, given the average number of landmarks each condition displayed.

#### *Navigation*

The navigation analysis also revealed some interesting characteristics of user navigation behaviour regarding the use of zooming and panning tools. In some conditions, participants favoured one type of navigation over the other—conditions with a higher average number of landmarks tended to favour zooming, and the inverse holds true for panning; but there was a wide variance between participants' navigation style.

It is unknown if the zooming technique had an impact on this. The zooming technique used performed a zoom on the point under the mouse cursor, rather than centring this view on the point under the cursor and zooming on that.

### 4.3.1 Experimental Concerns

#### *Organisation of Items*

Items were organised alphabetically in a grid. However, this was not the only possible configuration that could have been used. The grid configuration was chosen because it does not place a visual search bias in any one direction (studies by McMichael and McCarthy (1975) and White (1989) found visual search to be faster in a horizontal direction than in a vertical one) and gives the smallest number of zooming steps and shortest average path between any pair of items.

An issue observed with this configuration was with the continuous nature of the data—word ranges would often wrap across rows of the grid; a landmark would indicate the start of a word range related to the target selection at the end of a row, but the target would exist at the start of the following row. This caused frustration for participants, as they had to zoom and pan over to the following row and begin their visual search task again.

### *Predictable Data*

The use of alphabetic data allowed participants to predict the locations of items in the absence of landmarks to guide them—we believe this was the reason for the performance observed in the zero landmarks condition. Techniques—such as randomising the order of grid cells—would have reduced this effect, but at the detriment of the landmark's descriptive power; landmarks would no longer indicate the start of a contiguous group of data, but only a partial set.

### *Obstruction*

A complaint from participants was the obstruction of words by landmarks. Landmarks were always of a fixed visual size, and long landmarks often obscured words at distant zoom levels. Continuing to zoom in would have eliminated this effect, but participants were often unwilling to zoom more than necessary in order to be able to read the items, as shown in figure 4.5. Techniques—such as using transparent landmarks, or landmarks that reacted to the cursor position—would have reduced this frustration and error rate, but potentially decreasing performance (Harrison and Vicente (1996) found transparent menus and tools to have significantly reduced performance than opaque equivalents).

## 4.4 CONCLUSIONS

In this chapter, we have conducted an empirical evaluation of the landmarks principle from chapter 3. The results of our evaluation indicate strong support for our principle in both empirical results and subjective user evaluation.

CHAPTER 5

# Discussion & Conclusions

From the classification hierarchy of command interfaces presented in chapter 2, we have developed a set of principles relating to a branch of the hierarchy—navigation—in chapter 3, and evaluated one of the principles in chapter 4. In this chapter, we discuss the implications of our work on interface design, and the possibilities for future work.

## 5.1 LANDMARKS

The evaluation of our landmarking principle presented in chapter 4 has strong implications for interfaces that use landmarks to aid user navigation. Interfaces that necessitate the use of landmarking by virtue of the nature of their data, are the primary benefactors from this principle and evaluation. For example, in mapping interfaces—such as Google Maps—the relationships between sibling items are less predictable than those in set of alphabetic words. In such interfaces, the ability to identify these relationships is dependant on the user's knowledge of the data (although navigation tools, and features of navigation views can assist with this), and landmarks are required to provide rapid orientation. No landmarks requires a comprehensive visual search of low-level items, and too many landmarks induce further serial hunting tasks due to the visual clutter.

The primary goal of landmarks is to provide navigation aids at a density that assists the user without becoming distractors and inducing additional visual search tasks. The empirical and subjective results from our evaluation show that the logarithmic relationship principle has best supported this goal.

### 5.1.1 Future Work

An issue not considered in our evaluation was landmark selection—which landmarks to show to the user. In our evaluation, landmarks were chosen at random to avoid bias, but in production interfaces, this is not a viable option. Selection of landmarks that gave preference to the importance of the data that each landmark is representative of, or the frequency/recency of items used within the group are possible strategies that would result in more valuable landmarking. We believe that the selection of such a strategy is dependant on the data being shown, but this would benefit from future evaluation and comparison of such strategies.

The landmarks principle is still in its infancy and further evaluation of its application is required in order to establish its strength and validity. Evaluation in scenarios where landmark location would not carry such high predictive power as that in our evaluation should be conducted, as should the influence of different landmarking strategies.

We believe there may also an interesting interaction between landmarking and spatial memory principles. As landmarks are promoted as spatially stable identifiers and can be rapidly searched, there is reason to believe they may improve development of spatial memory of the data being navigated.

We also believe there may be a connection to the Hick-Hyman Law, reviewed in section 2.4.1. Due to the low information content of landmarks (probability varies with the landmark selection strategy), landmarks would appear to have a short reaction time—supported by the Hick-Hyman Law and our evaluation results. Future work may investigate the link and possible model for human performance between our developing model for landmark visibility and the Hick-Hyman Law.

Irrespective of the Hick-Hyman Law, we are interested in future work that further establishes our principle for landmarking as an accurate model that can be applied by designers of navigation interfaces.

## 5.2 NAVIGATION AND COMMAND INTERFACES

The principle for landmark visibility is only one of those presented in chapter 3 that influences user interaction with navigation interfaces. We believe that our principles and definition of navigation interfaces accurately describes the construction of, and influences on, navigation interfaces. These are the principles that are relevant to designers in the construction of future interfaces.

Due to time constraints, only one of the principles presented was evaluated. Support already exists for several of the principles outlined, but we believe the model would benefit from future work to establish the validity of the principles and—just as importantly—the interactions between them.

The navigation principles themselves are part of a larger classification of command interfaces, presented in chapter 2. The same study of navigation interfaces can be conducted on the three other top-level categories in our hierarchy. We are especially interested in future work that describes the interactions between categories within the hierarchy. We believe that future development of this classification hierarchy can lead to new models for describing user interaction with command interfaces.

## 5.3 CONCLUSIONS

We have explored the theory behind user interaction with command interface, and conducted an in-depth analysis of the navigation components of command interfaces, resulting in an empirical evaluation of a specific principle. Results of the evaluation have shown our principle accurately describes the number of landmarks that should be presented to the user for maximum efficiency.

The evaluation conducted into landmarks has supported our claims and supports development into a model for describing human performance with landmarks. Our claim for a logarithmic relationship between the number of visible items in an interface and the number of visible landmarks was compared against several other relationships, performing significantly better than the other landmarking conditions. The experimental results also support our claims for landmarks that do not increase the mental load on the user when conducting a visual search, and the ability for our principle to support this goal.

Our principles for navigation draw on a large body of research on interfaces and human performance, but they also present new principles and interactions between them that have not been explored by prior work.

In this study we have built a framework for describing the influences of command interfaces on human performance, we have developed part of this framework into a set of principles oriented towards navigation interfaces, and successfully evaluated one of the principles—showing its impact on human performance. We believe that this principle, and future development of our our classification framework will lead to better insight and understanding into the foundational influences on user performance.

# Bibliography


Accot, J. and Zhai, S. (1997). Beyond Fitts' law: models for trajectory-based HCI tasks, *CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 295–302.

Accot, J. and Zhai, S. (1999). Performance evaluation of input devices in trajectory-based tasks: an application of the steering law, *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 466–472.

Au, I. and Li, S. (1998). Netscape communicator's collapsible toolbars, *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, pp. 81–86.

Beaudouin-Lafon, M. (2000). Instrumental interaction: an interaction model for designing post-WIMP user interfaces, *CHI '00: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 446–453.

Beck, J. (1966). Effect of orientation and of shape similarity on perceptual grouping, *Perception & Psychophysics* **1**(9): 300–302.

Bederson, B. B. and Boltman, A. (1999). Does animation help users build mental maps of spatial information?, *Proceedings of the 1999 IEEE Symposium on Information Visualization*, pp. 28–35.

Bier, E. A., Stone, M. C., Fishkin, K., Buxton, W. and Baudel, T. (1994). A taxonomy of see-through tools, *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 358–364.

Cao, X. and Zhai, S. (2007). Modeling human performance of pen stroke gestures, *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 1495–1504.

Card, S. K., Thomas, T. P. and Newall, A. (1983). *The Psychology of Human-Computer Interaction*, Lawrence Erbaum Associates, London.

Chapanis, A. and Lindenbaum, L. E. (1959). A reaction time study of four control-display linkages, *Human Factors* **1**(4): 1–7.

Chignell, M. H. (1990). A taxonomy of user interface terminology, *SIGCHI Bull.* **21**(4): 27.

Cockburn, A., Gutwin, C. and Alexander, J. (2006). Faster document navigation with space-filling thumbnails, *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, ACM Press, New York, NY, USA, pp. 1–10.

Cockburn, A., Gutwin, C. and Greenberg, S. (2007). A Predictive Model of Menu Performance, *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA.

Cockburn, A., Karlson, A. and Bederson, B. B. (n.d.). A Review of Overview+Detail, Zooming, and Focus+Context Interfaces. In submission to ACM Computing Surveys.

Cockburn, A., Savage, J. and Wallace, A. (2005). Tuning and testing scrolling interfaces that automatically zoom, *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, New York, NY, USA, pp. 71–80.

Coren, S. and Girgus, J. S. (1980). Principles of perceptual organization and spatial distortion: The gestalt illusions, *Journal of Experimental Psychology: Human Perception and Performance* **6**(3): 404–412.

de Alwis, B. and Murphy, G. C. (2006). Using Visual Momentum to Explain Disorientation in the Eclipse IDE, *IEEE Symposium on Visual Languages and Human-Centric Computing 2006*, pp. 51–54.

Findlater, L. and McGrenere, J. (2004). A comparison of static, adaptive, and adaptable menus, *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 89–96.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement., *Journal of Experimental Psychology* **47**(6): 381–391.

Furnas, G. W. (1986). Generalized fisheye views, *CHI '86: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 16–23.

Furnas, G. W. (1997a). Effective view navigation, *CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 367–374.

Furnas, G. W. (1997b). Effective view navigation. Available at URL.
**URL:** *http://www.si.umich.edu/~furnas/Papers/EVN.HCIC95.workshop.paper.pdf*

Furnas, G. W. (2006). A fisheye follow-up: further reflections on focus + context, *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, ACM Press, New York, NY, USA, pp. 999–1008.

Gajos, K. Z., Czerwinski, M., Tan, D. S. and Weld, D. S. (2006). Exploring the design space for adaptive graphical user interfaces, *AVI '06: Proceedings of the working conference on Advanced visual interfaces*, ACM Press, New York, NY, USA, pp. 201–208.

Goldberg, D. and Richardson, C. (1993). Touch-typing with a stylus, *CHI '93: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 80–87.

Gonzales, J. I. B. (1994). A theory of organization, *SIGDOC '94: Proceedings of the 12th annual international conference on Systems documentation*, ACM Press, New York, NY, USA, pp. 145–155.

Greenberg, S. and Witten, I. H. (1993a). Supporting command reuse: empirical foundations and principles, *International Journal of Man-Machine Studies* **39**(3): 353–390.

Greenberg, S. and Witten, I. H. (1993b). Supporting command reuse: mechanisms for reuse, *International Journal of Man-Machine Studies* **39**(3): 391–425.

Hanson, S. J., Kraut, R. E. and Farber, J. M. (1984). Interface design and multivariate analysis of UNIX command use, *ACM Trans. Inf. Syst.* **2**(1): 42–57.

Harrison, B. L. and Vicente, K. J. (1996). An experimental evaluation of transparent menu usage, *CHI '96: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 391–398.

Hick, W. E. (1952). On the rate of gain of information, *Quarterly Journal of Experimental Psychology* **4**: 11–26.

Hornbæk, K., Bederson, B. B. and Plaisant, C. (2002). Navigation patterns and usability of zoomable user interfaces with and without an overview, *ACM Trans. Comput.-Hum. Interact.* **9**(4): 362–389.

Hornbæk, K. and Frøkjær, E. (2003). Reading patterns and usability in visualizations of electronic documents, *ACM Trans. Comput.-Hum. Interact.* **10**(2): 119–149.

Hornof, A. J. (2001). Visual search and mouse-pointing in labeled versus unlabeled two-dimensional visual hierarchies, *ACM Trans. Comput.-Hum. Interact.* **8**(3): 171–197.

Hyman, R. (1953). Stimulus information as a determinant of reaction time, *Journal of Experimental Psychology* **45**(3): 188–196.

Jul, S. and Furnas, G. W. (1997). Navigation in electronic worlds: a CHI 97 workshop, *SIGCHI Bulletin* **29**(4): 44–49.

Kahneman, D. and Henik, A. (1981). Perceptual Organization and Attention, *in* M. Kubovy and J. R. Pomerantz (eds), *Perceptual Organization*, Lawrence Erlbaum Associates, Hillsdale, New Jersey, pp. 181–211.

Kristensson, P.-O. and Zhai, S. (2005). Relaxing stylus typing precision by geometric pattern matching, *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, ACM Press, New York, NY, USA, pp. 151–158.

Kurtenbach, G. and Buxton, W. (1994). User learning and performance with marking menus, *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 258–264.

Landauer, T. K. and Nachbar, D. W. (1985). Selection from alphabetic and numeric menu trees using a touch screen: breadth, depth, and width, *CHI '85: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 73–78.

Lee, E., Whalen, T., McEwen, S. and Latrémouille, S. (1984). Optimizing the design of menu pages for information retrieval, *Ergonomics* **27**(10): 1051–1069.

MacLean, A., Carter, K., Lövstrand, L. and Moran, T. (1990). User-tailorable systems: pressing the issues with buttons, *CHI '90: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 175–182.

McMichael, E. and McCarthy, S. V. (1975). Visual search through words and nonwords in horizontal and vertical orientations, *Perceptual and Motor Skills* **41**(3): 740–742.

Miller, G. A. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information, *The Psychological Review* **63**: 81–97.

Nigay, L. and Vernier, F. (1998). Design method of interaction techniques for large information spaces, *AVI '98: Proceedings of the working conference on Advanced visual interfaces*, ACM Press, New York, NY, USA, pp. 37–46.

Norman, D. A. (1983). Some Observations on Mental Models, *in* D. Gentner and A. L. Stevens (eds), *Mental Models*, Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey.

Norman, D. A. (1986). Cognitive Engineering, *in* D. A. Norman and S. W. Draper (eds), *User Centered System Design*, Lawrence Erlbaum Associates, Inc., Hillsdale, New Jersey, pp. 31–61.

Norman, D. A. (1988). *The Design of Everyday Things*, Doubleday, New York, New York.

Norman, K. L., Weldon, L. J. and Shneiderman, B. (1986). Cognitive layouts of windows and multiple screens for user interfaces, *International Journal of Man-Machine Studies* **25**(2): 229–248.

Page, S. R., Johnsgard, T. J., Albert, U. and Allen, C. D. (1996). User customization of a word processor, *CHI '96: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 340–346.

Pastel, R. L. (2006). Measuring the Difficulty of Steering Through Corners, *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, ACM Press, New York, NY, USA, pp. 1087–1096.

Perlin, K. and Fox, D. (1993). Pad: an alternative approach to the computer interface, *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, pp. 57–64.

Pirolli, P. (1997). Computational models of information scent-following in a very large browsable text collection, *CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press, New York, NY, USA, pp. 3–10.

Pirolli, P. and Card, S. (1999). Information foraging, *Psychological Review* **106**(4): 643–675.

Plaisant, C., Carr, D. and Shneiderman, B. (1995). Image-Browser Taxonomy and Guidelines for Designers, *IEEE Software* **12**(2): 21–32.

Pomerantz, J. R. (1981). Perceptual Organization in Information Processing, *in* M. Kubovy and J. R. Pomerantz (eds), *Perceptual Organization*, Lawrence Erlbaum Associates, Hillsdale, New Jersey, pp. 141–180.

Rock, I. and Palmer, S. (1990). The legacy of Gestalt psychology, *Scientific American* **263**(6): 84–90.

Roske-Hofstrand, R. J. and Paap, K. R. (1986). Cognitive networks as a guide to menu organization: An application in the automated cockpit, *Ergonomics* **29**(11): 1301–1311.

Sears, A. and Shneiderman, B. (1994). Split menus: effectively using selection frequency to organize menus, *ACM Trans. Comput.-Hum. Interact.* **1**(1): 27–51.

Stroop, J. R. (1935). Studies of interference in serial verbal reactions, *Journal of Experimental Psychology* **18**(6): 643–662.

Stuerzlinger, W., Chapuis, O., Phillips, D. and Roussel, N. (2006). User interface façades: towards fully adaptable user interfaces, *UIST '06: Proceedings of the 19th annual ACM symposium on User interface software and technology*, ACM Press, New York, NY, USA, pp. 309–318.

Tang, K.-H. E. (2001). Menu Design with Visual Momentum for Compact Smart Products, *Human Factors* **43**(2): 267–277.

Treisman, A. and Gelade, G. (1980). A feature-integration theory of attention, *Cognitive Psychology* **12**(1): 97–136.

Wallace, A., Savage, J. and Cockburn, A. (2004). Rapid visual flow: how fast is too fast?, *AUIC '04: Proceedings of the fifth conference on Australasian user interface*, Australian Computer Society, Inc., Darlinghurst, Australia, Australia, pp. 117–122.

Waterworth, J. and Chignell, M. (1991). A Model of Information Exploration, *Hypermedia* **3**(1): 35–58.

Wertheimer, M. (1958). Principles of Perceptual Organization, *in* D. C. Beardslee and M. Wertheimer (eds), *Readings in Perception*, D. Van Norstrand Company, Inc., New York, pp. 115–135. An abridged translation by Michael Wertheimer of Untersuchungen zur Lehre von der Gestalt, II. *Psychol. Forsch.*, 1923, 4, 301–250.

White, M. J. (1989). Effect of calendar layout on calendar search, *Ergonomics* **32**(1): 15–25.

Woods, D. D. (1984). Visual momentum: a concept to improve the cognitive coupling of person and computer, *International Journal of Man-Machine Studies* **21**(3): 229–244.

Zhai, S., Accot, J. and Woltjer, R. (2004). Human Action Laws in Electronic Virtual Worlds: An Empirical Study of Path Steering Performance in VR, *Presence: Teleoperators and Virtual Environments* **13**(2): 113–127.

Zipf, G. (1949). *Human behavior and the principle of least effort: An introduction to human ecology*, Addison-Wesley Press, Oxford, England.