Breaking up the Kanji

A conceptual electronic dictionary design based upon the cognitive sub-character reality of Chinese characters

Author:
Jeroen Douwe Hoek
✉ mail@jeroenhoek.nl

Leiden University
Department of Japanese Studies

Dissertation submitted in partial fulfilment of the requirements for the degree Master of Arts *in* Japanese Studies

19th August 2009

The body of this document is typeset using *Gentium* (12pt) for normal Roman text, *DejaVu Sans Mono* for monospaced Roman text, *Hanazono Minchō* for Chinese characters, radicals, and Chinese character components, and *IPA Mona Minchō* for katakana and hiragana text. The title and subtitle are typeset in *Open Din Schriften Engschrift* and *DejaVu Sans ExtraLight*, respectively.

This document was produced with the LaTeX document preparation system. All illustrations were created with the use of Inkscape and the GIMP.

All software and typefaces used for the creation of this work are free software.

Abstract

Japanese is by no means an easy language to learn, due in part to the many kanji (Chinese characters) it uses. In this research I propose a new character and word look-up concept for use in electronic dictionary software specialised in dealing with the Japanese language. By taking the cognitive processing reality of kanji into account, I have designed a graphical user interface concept that allows the user to interact with kanji and its constituent parts on a sub-character level, and to perform more intuitive search queries against existing digital kanji and word dictionaries. The key component of this design is the *kanji navigator*; a conceptual widget that will allow users to do this. This thesis focuses mainly on second language (L2) learners of Japanese from a non-kanji cultural background.

抄録

日本語の学習は、ある程度漢字総数の大きさゆえに、決して簡単なことではない。この研究では、日本語対応できる電子辞書ソフトウェアのため、新たな文字・語への検索方法論を考察したいと思う。漢字の認知的処理の現実を踏まえ、利用者が文字内レベルで漢字とその漢字の部品と対話できるグラフィカルユーザインタフェース概念を設計した。さらに、そのインタフェースで、既存のデジタル漢字や日本語などの辞書への検索も、より直覚的にできるようなものである。このデザインの主要な要素とは、上記の機能を可能させる概念的なウィジェット、いわゆる「漢字ナビゲーター」である。この論文では、利用者は主に第二言語、非漢字圏出身の日本語学習者を対象にした。

Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

Learning any language anew takes considerable effort, but some languages are, arguably, harder to acquire than others. Japanese in particular has a reputation for being a *challenging* language, partly because of its complex writing system. For learners hailing from a culture where the logographic Chinese characters are not a part of daily life, the study of Japanese as a second language (L2) can be quite difficult, due to this unfamiliarity with the numerous *Chinese characters*[1], or *kanji*, used in Japanese. Learning to read and write hundreds of complicated characters is a daunting task, to say the least.

Fortunately, the learner of Japanese does not have to face this challenge alone; his trusty aides are the dedicated Japanese language teachers, the instructive text books, and of course that one true friend that remains at his side when teachers are no longer available and text books have nothing left to teach you, the humble *dictionary*. However, using a dictionary for the Japanese language (such as a kanji dictionary or a Japanese–English dictionary) can be a time-consuming process due to the nature of the Japanese language — or more precisely, due to the *kanji*.

[1] The term "Chinese character" is slightly ambiguous; in this thesis it refers to the logographic characters that originated in China, and are (or were) used in *all* the China, Japan, Korea and Vietnam (CJKV) cultures. Some sources opt to use the term "Han character" instead. *Kanji* is the Japanese term for the same concept, but in this thesis is used when talking about Chinese characters within the context of the Japanese language.

1.1 The Japanese language

The modern written Japanese language uses four distinct scripts; two syllabaries, the Roman alphabet, and kanji. These scripts each serve their specific purposes and are used in mixed sentences (*majiribun*) such as the one in illustration 1.1[2].

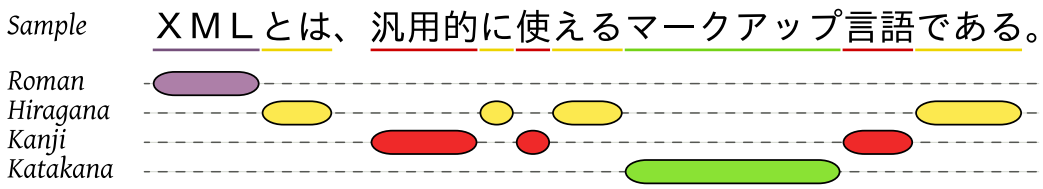


Illustration 1.1: Japanese *majiribun* example featuring all four scripts

The two *kana* syllabaries called *hiragana* and *katakana* are, on the whole, not very difficult to master. Basically, the two syllabaries use different characters (46 each) for the same set of sounds — or more precisely, morae. All of the sounds in the Japanese language can be represented using either of the kana syllabaries. Broadly speaking, in terms of functionality hiragana could be called the glue of the Japanese written language, and is used for verb, adjective and adverb inflections (*okurigana*), grammatical particles, occasionally personal names[3], and for words that can be written using kanji, but are not because the kanji are obscure (and the target audience is not expected to know them) or because the word is customarily written exclusively in hiragana. Similarly, hiragana can also be used to indicate the reading for words written in kanji, by placing the reading in hiragana above the kanji word — such kana are called *furigana* or *ruby* (Nelson & Haig 1997, p. 1247). The katakana syllabary is customarily used for foreign loanwords, foreign names, or emphasis. The Roman alphabet is mainly used for abbreviations and poses no problem for most learners of Japanese.

[2] Transcribed using the Hepburn romanisation system, the sample sentence reads: *XML to wa, han'yōteki ni tsukaeru mākuappu gengo de aru* ("XML is a multi-purpose mark-up language").

[3] Although most Japanese names are written in kanji, children are sometimes given first names written in hiragana.

1.1.1 Kanji and Japanese

While hiragana may be the glue that holds Japanese sentences together, the kanji are the bones. Nouns, verb, adjective and adverb stems, names, etcetera, are usually written using kanji. Kanji were introduced into the written language during several periods of cultural borrowing from various Chinese dynasties. One oft-heard question any L2 learner of Japanese will have been posed by friends or family at some point, is about the actual number of kanji you need to know to be able to read and write in Japanese. "It depends", is perhaps the most common answer, often followed by "roughly two or three thousand"[4]. A common guideline for basic literacy is the *Jōyō Kanji* (common use kanji) set, defined by the Japanese Ministry of Education, which currently[5] lists 1945 kanji deemed "most common" in the Japanese language. A subset of this chart called *Kyōiku Kanji* (education kanji) comprised of 1006 kanji is taught in elementary and junior high school over the course of six years. High school students are expected to have mastered the remaining kanji listed in the Jōyō Kanji chart upon graduation.

However, reading a Japanese newspaper, enjoying Japanese literature, or even indulging oneself in one of Japan's popular *manga* (comic) titles more often than not implies encountering kanji not part of the Jōyō Kanji chart. Kanji frequency studies based on newspaper archives suggest that 3000 kanji cover almost all (99%) of what Japanese readers may encounter, but that the remaining cases are spread out over another 3000 kanji (Kess & Miyamoto 1999, p. 199; citing Nozaki, et al. 1996). Similarly, Hadamitzky & Spahn (1997, p. 43) estimate that roughly 6000–7000 kanji are used in modern Japanese, and that Japanese of average education are familiar with about 3000 kanji.

How many kanji you need to master — or at least be able to read and comprehend — to achieve functional fluency in Japanese, varies greatly depending on the subject field as well as individual needs and interests. A software developer may find that most of the text he encounters and produces certainly contain

[4] Or more vaguely put, "A lot, but thankfully not as much as Chinese".

[5] The Jōyō Kanji chart, which superseded the smaller *Tōyō Kanji* (daily use kanji) chart in 1981, is currently up for revision. The Agency for Cultural Affairs is expected to present their final proposal for the new official Jōyō Kanji chart in autumn 2010. Among the new kanji and readings in the list are those used in the names of Japan's prefectures. At the time of writing the new list is expected to total 2131 kanji.

plenty of jargon, but that the words he uses are often English loanwords rendered in the phonetic *katakana* syllabary, and that the kanji used are predominantly part of the Jōyō Kanji chart. Illustration 1.2 shows a passage from this type of text[6].

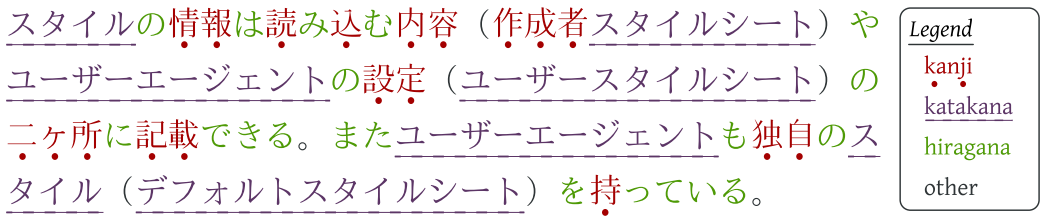


Illustration 1.2: Japanese text sample with many katakana words

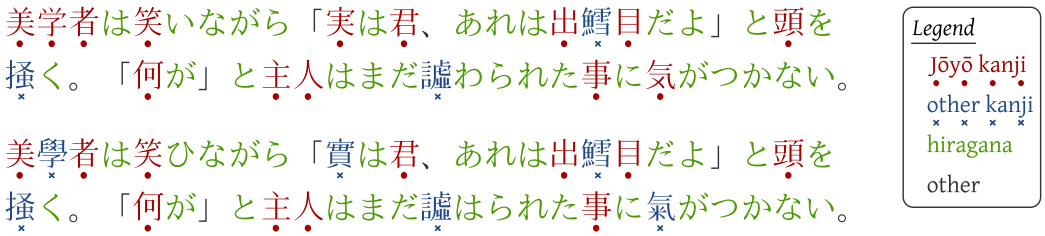


Illustration 1.3: Fragment from *I am a Cat*, showing rare kanji usage

All kanji in illustration 1.2 can be found in the Jōyō Kanji chart, and all of the kanji words are fairly common. Contrast this with the following snippet[7] from Meiji-era author Natsume Sōseki's celebrated novel *I am a Cat*, in illustration 1.3. The same sentence is repeated twice; once written according to modern-day kanji usage conventions, and below it the original. Even in the modern rendition

[6] A note on style sheet handling by web browsers, taken from an article on Cascading StyleSheets (CSS) found in the Japanese Wikipedia, accessed on 3rd May 2009. The text reads:

> Style information can be declared in two locations; in the fetched content (the author's stylesheet) and in the settings of the user-agent (the user's stylesheet). The user-agent provides its own style (the default stylesheet) as well.

[7] The eponymous cat observes a conversation its owner is having. Roughly translated the sentence reads:

> Laughing, the aesthete scratched his head; "actually, I made that up". "Made what up?", my master asked, not yet realising he had been deceived.

of this text already there are three characters used which are not part of the Jōyō Kanji chart, and in the original passage yet another three uncommon kanji[8] appear. In contrast with the earlier loanword-laden sample, there are no katakana words, and a lot of the words used are either uncommon, or even quite rare in the appearance chosen by the author — he chose to use the form 譸つる for the verb normally written as 偽る (**itsuwaru**, *to deceive*), and the word 出鱈目 (**detarame**, *nonsense*) is now only rarely written in kanji.

Granted, the lower sentence in illustration 1.3 is an extreme case; the work was published in 1905, and as such uses kanji that are no longer in common use[9], but it serves to illustrate that there is no definitive set of kanji that can be considered indispensable, as this is a highly subjective matter. Regardless of one's field of interest, encountering unfamiliar kanji is simply part and parcel of learning and using Japanese. Everybody, from the novice learner to the seasoned scholar, will have to deal with unknown characters from time to time. Place names and historical names in particular are frequently written in kanji that may not be in common use.

1.2 Dictionaries

When confronted with the task of creating a dictionary for a language with thousands of unique characters, the notion of simply listing all entries in some sort of alphabetical order no longer applies. Of course, the various countries that use Chinese characters in their language do have methods of ordering the characters they use — for example, by using the character reading and sorting according to its representation in the Roman alphabet or in hiragana, or by using the order already used by some renowned classical dictionary, a character encoding standard, or a government issued list such as the Jōyō Kanji chart — but it is infeasible for a normal human being to learn such a sequence of thousands by heart in the way this is done with the twenty-odd letters of the Roman alphabet.

[8] Older forms (*kyūjitai*) of 学, 実and 気(學, 實and 氣, respectively).

[9] In 1946 the Japanese government enacted a series of script reforms simplifying a number of kanji, sometimes drastically altering their shape (for example, 發became 発).

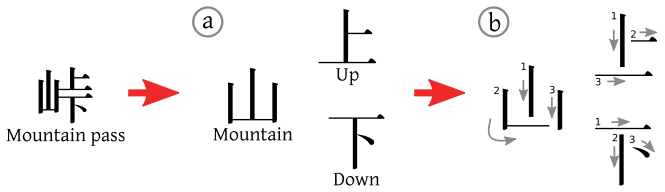


Illustration 1.4: The kanji for "mountain pass" (*tōge*) split up into its components (a) and strokes (b)

Individual kanji may be unique, but the vast majority of kanji are made up of a number of recurring components (Tamaoka & Yamada 2000, p. 199), often kanji in their own right. For example, illustration 1.4 shows the decomposition for the kanji 峠 Utilising this characteristic, the Chinese overcame the problem of kanji indexing by introducing the system of 214 radicals; part of a Chinese character was designated its *radical*. This radical acted as a rough semantical classifier; for instance, characters with the radical *wood/tree* 木have a meaning that is related to wood or trees, wooden objects, or concepts that relate to properties of wood; in short, the radical gives a rough indication of the lexical domain (Saito et al. 1998, p. 325). Still, splitting up a large set of Chinese characters into 214 categories may leave us with smaller, but nonetheless unwieldy groups of characters, so another indexing criteria is needed. During that delicate period when the new learner of Japanese grapples with the basics of writing kanji, many a teacher and textbook will have stressed the importance of writing the characters using their correct stroke count and order. One reason for this is that it helps us write characters that are — to some extent, in any case — legible, but it also has the benefit of teaching us how to count the number of strokes in a Chinese character. The radical indexing system leverages this know-how, by employing a two-prong approach. First, identify the radical of the character. Secondly, count the number of strokes found in the remaining components. The combination of these two techniques limits the list of potential candidates to a manageable tens, rather than hundreds or thousands of characters, so all we need to do is memorise the most common of these 214 classical radicals (*The New Nelson* (Nelson & Haig 1997) lists 67 of the most frequently encountered radicals as "more important"), and learn to identify a Chinese character's radical.

Unfortunately, this is not always as easy as it sounds, not to mention time consuming. Although there are guidelines for identifying a Chinese character's radical by taking its position in the character into consideration, many exceptions and ambiguous cases exist. Table 1.1 shows ten kanji that all share a similar composition; the 門(*gate*) component surrounding another component. If we follow the traditional radical identification guidelines, 門should be the radical for all of these kanji, but a number of them behave differently.

	Regular							Irregular		
Kanji	門	間	閣	閑	関	開	閉	聞	悶	問
Radical	門	門	門	門	門	門	門	耳	心	口

Table 1.1: Kanji where following the traditional radical identification guidelines, 門 might be expected as radical

When the concept of radicals was brought into practice many centuries ago, the radical was introduced as the component of the character that carried its meaning. Often, following the guidelines listed in most kanji dictionaries, the radical can be identified, but not always. For instance, in table 1.1 the radical for the character meaning "to hear" (聞) is 耳, because that is also the kanji for "ear", and thus signifies a semantic classification of the whole (Nelson & Haig 1997, p. 1233). This system may have made perfect sense to the Chinese scholars that invented it, but in modern usage the radical of a kanji does not always make sense as the carrier of (rough) semantic meaning, and feels counter-intuitive (Saito et al. 1995*a*, p. 115). Moreover, using a dictionary usually implies that we are trying to find the meaning of a character in the first place (Hadamitzky & Spahn 1997, p. 65), so as a method of finding kanji in a dictionary there is room for improvement.

Paper Japanese kanji dictionaries have endeavored to make finding kanji easier by adding supplementary indices; such as an index that lists kanji ordered by reading, but this requires the user to know the reading beforehand. Even if we do know the reading, for more common readings such as "kō" or "shō" the list of candidates is quite long. In *The New Nelson* the so-called *universal radical index* (Nelson & Haig 1997, appendix 15) was introduced. This index can be used with

a wrongly identified radical and the remaining stroke count. For example, the kanji 聞may not be found under the 門radical heading in the main body of dictionary, but we could retrieve it by looking in this extra index. Although this approach alleviates some of the issues with the traditional radical and stroke count method, the drawback is that this constitutes an additional operation for the user, adding to what is already a time consuming process.

Alternatives to the radical and stroke system exist, but few appear in widespread use today. One notable alternative indexing system is SKIP (System of Kanji Indexing by Patterns) introduced in Jack Halpern's *Kanji learner's dictionary* published by Kodansha. The SKIP method uses a combination of kanji composition and stroke count to calculate a three number code that can be used to look up the kanji in a dictionary with such an index. The only requirement to the user is the ability to count strokes — which requires a little training, but should not pose a problem to anyone learning to read and write kanji — and to classify the kanji as one of four possible classes, which becomes the first number of the SKIP code: left-right (1), up-down (2), enclosure (3), and solid (4). For example, the kanji 峠(mountain pass) can be split vertically into 山and 卡, so the first number will be **1**; the stroke counts of the remaining components (the left and right component counted as **3** and **6** respectively) are then used for the two remaining numbers, giving us the SKIP code **1-3-6**. The idea of this system is to allow novice learners of Japanese to find kanji without intimate knowledge of the traditional radicals, using a only limited number of rules. Potential drawbacks are the time and effort required to calculate the SKIP code for kanji; often the exact stroke count for the more difficult kanji is also harder to count.

1.2.1 Electronic dictionaries

With the spread of personal computers and the introduction of the first portable electronic dictionaries (PEDs) for the Japanese language — known as *denshi jisho* — in 1991 (Ishikawa 2004), electronic dictionaries made their debut. The digitalisation of dictionaries brought along a number of practical advantages; dictionary look-ups could now be performed in a fraction of second as long as the

reading of the word or kanji was known, and if used on a personal digital assistant (PDA), smartphone, netbook or as a PED, they were now also portable.

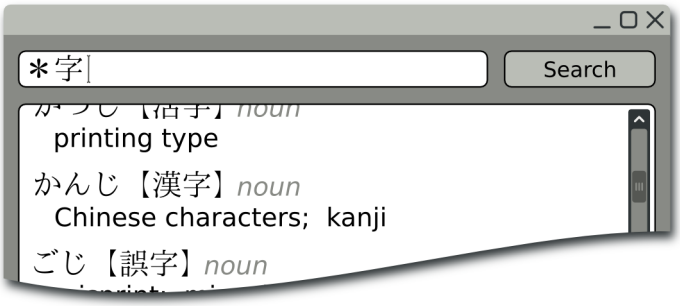


Illustration 1.5: Basic dictionary interface showing a query using wildcards

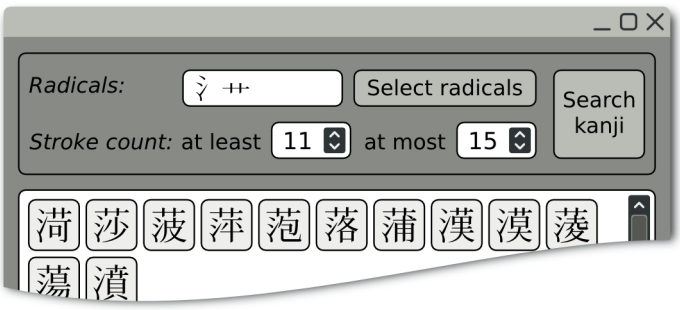


Illustration 1.6: Kanji search screen with radical and stroke count arguments entered

Initially, in an era where typewriters were rapidly becoming indispensable in offices around the world, the Japanese script proved difficult to deal with; office automation in Japan was held back for years due to the absence of an intuitive method to type Japanese text on a typewriter (Gottlieb 2000). This issue was eventually overcome with the introduction of the well-known concept and widespread implementation of the input method editor (IME) for the Japanese language. Now, as long as we know the exact pronunciation, we can enter most of the Japanese vocabulary without significant slowdowns.

The most basic form of electronic dictionary software can be used for almost any language, and simply provides the user with a single text input widget and a scrollable results area (illustration 1.5). The inputting of words in Japanese is

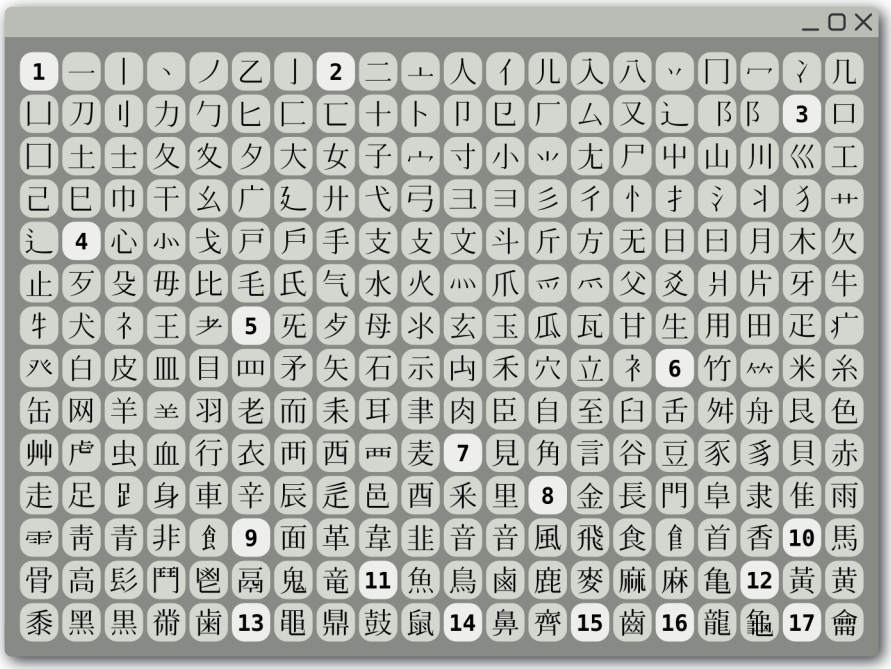


Illustration 1.7: Typical radical selection widget showing the 214 traditional radicals and their variants

simply delegated to the IME provided by the operating system (OS)[10]. Such electronic dictionary applications often allow for the use of wildcards. For example, illustration 1.5 shows a query asking for all words ending in 字. The *asterisk* stands for "zero or more characters". Similarly, some dictionary applications allow for the use of a *question mark* in the query as well, signifying "any one character". This is useful if the user knows how to input one kanji of, say, a two kanji compound word.

In electronic dictionary software too, the radical and stroke count method is widely implemented; usually in a manner similar to illustration 1.6. The user can select the desired radicals from a grid listing the 214 classical radicals and its variants, grouped by stroke count (illustration 1.7). As with the universal radical index in *The New Nelson*, any radical that is part of the kanji can be used,

[10] In the case of PEDs, the IME is built into the device; from a user's perspective the OS, IME and electronic dictionary software on these devices appear inseparable.

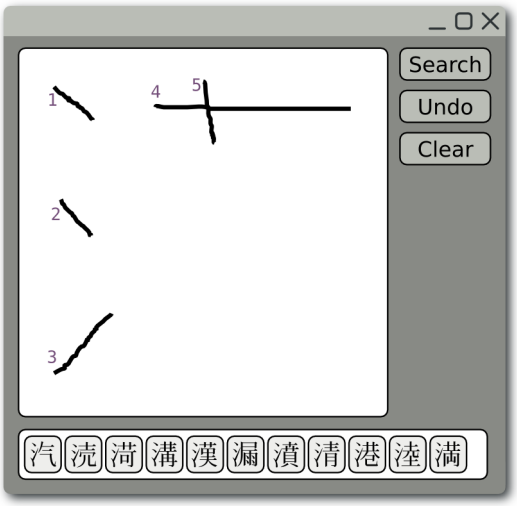


Illustration 1.8: Kanji drawing pad showing an attempt to enter the kanji 漢and enough strokes drawn in to find it

regardless of whether or not it actually is its proper radical. Additionally, the stroke count of the kanji can be used to further limit the search results.

Finally, IMEs and the more upmarket PEDs come equipped with kanji drawing functionality as an alternative means of input (illustration 1.8). By using a pointing device, such as a mouse or a stylus, strokes drawn by the user are algorithmically recognised by the software. Kanji matching the strokes input are suggested to the user on-the-fly; often a few strokes are sufficient to find the desired character in the suggestion list. This method does require that the user observes the correct stroke order and stroke count, as well as the location of the strokes, making it ill-suited for cases where the user does not know the exact strokes to use, is mistaken about the stroke order, or unable to correctly discern where strokes begin or end.

1.3 Limitations of current dictionary designs

A problem with most kanji dictionary software is that they appear to rely on the assumption that the user already knows quite a lot about the character he is looking for (Apel & Quint 2004). If the reading is known, he can use an index

that lists kanji ordered by reading, or in digital environment, the IME can display a list of applicable candidates ordered by a weighted average of commonness and personal usage of the word or kanji. If the radical and the stroke count are known, the method traditionally employed by paper dictionaries can be used. Applications that implement these techniques are widely available, either as stand-alone software, PEDs or as on-line dictionaries accessible with an internet browser.

In Hoek (2007) a number of problematic cases were identified where the currently available dictionary concepts are impractical, or could not be used at all. A possible solution considered was a method that allowed sub-character features to be used. At the word-level, most electronic dictionary applications treat kanji as indivisible blocks, as if dealing with an extraordinarily large alphabet, but kanji are not letters; most kanji consist of visually distinct components, such as radicals and other kanji. Humans do not look at kanji as monolithic blocks, but how do we process them? Can we provide a digital environment for dealing with kanji that capitalises on our cognitive capabilities?

1.4 Research aim

The goal of this research is to propose a blueprint for an electronic dictionary software application designed to allow users to search for kanji and words using a wider array of search criteria; in particular, by allowing the user to "interact" with the kanji on the sub-character level. The present research is but the first step in the development of the application proposed; eventually, a complete implementation of the design presented herein, will build upon this conceptual design, identifying faults and shortcomings, and dealing with the more complicated technical issues along the way.

By necessity, the nature of this research is interdisciplinary; cognitive linguistics and psycholinguistic research tell us how humans perceive kanji, the relatively young field of Human-Computer Interaction (HCI) provides some clues to the way humans interact with computers, and software engineering deals with the practical aspects of implementing the software. This thesis is divided accordingly, in essence answering the *why*, *what* and *how* of the software proposed.

In chapter 2 the cognitive reality of human kanji processing is explored. By understanding how we look at kanji, and how certain elements of these Chinese characters are treated in the cognitive process, a more intuitive and effective electronic dictionary application may be developed. This chapter does not represent a complete treatise of the differences between L1 and L2 learners, rather, it serves as the theoretical foundation for the design presented thereafter.

The point of interaction between man and computer is the graphical user interface (GUI); for an electronic dictionary application too, this is a (or perhaps *the*) critical part of the software. The toughest issue currently faced in the design of this class of software is not so much the *availability* of knowledge, the problem is how to make it accessible, and how to present only what is relevant. In chapter 3 the first step in the design process is outlined, and conceptual mock-ups for the various GUI concepts are presented.

Chapter 4 provides some clues as to how such a software application might be developed and implemented. The resources this software depends on — data such as databases with data on kanji composition and glyph shape definitions — are also introduced. Although the software proposed in this thesis is still in the planning stages at the time of writing, development of critical components is already underway. A free, open-source project housing these developments is presented alongside the technical implementation details in this chapter. Finally, in the concluding chapter possible directions for future developments are considered, and lacunae in the current research identified.

Chapters 3 and 4 are admittedly fairly light on details — mainly because a complete treatise of these subjects would fall well beyond the scope of an MA thesis. The main objective of the present research is to provide the *rationale* for the ideas presented.

Chapter 2

Humans and kanji: kanji from a cognitive perspective

For a proper analysis of the cognitive processes that occur when reading or writing kanji — and how understanding these processes might help us build a better electronic dictionary — we need to acknowledge that from a human perspective the Chinese characters used in the Japanese language are not treated in the same manner as letters from an alphabet. Whereas letters from, say, the Roman alphabet represent the smallest visual units for languages that use them, Chinese characters are often composed of distinct graphical components with specific features. These sub-character features in particular are of interest; kanji components appear to trigger phonetic and semantic processing, and they play a key-role in the mental lexicon. Could an increased focus on these components benefit L2 learners?

2.1 Defining "common kanji"

Before we explore the human perspective on kanji, the issue of how many kanji are in actual use in Japanese may need some addressing. In the introduction chapter (p. 3) we noted that there is no absolute measure for what kanji are, and are not needed for daily use, because this is a very subjective matter; a tenured professor in Japanese literature most likely has a wider repertoire of kanji he

uses or encounters on a daily basis than the average sushi chef[11]. But even for these two (hypothetical) extremes, there will be a large degree of overlap for the more common kanji.

In theory, the amount of kanji that can exist is infinite. Just as anyone can coin new words using the set of twenty-odd Roman letters used in most languages of European origin, it is also possible to construct new kanji using the common parts found in existing kanji. Of course, poetic as this may sound, this rarely — if ever — happens in modern Japanese, except for creative or artistic purposes. Nowadays new concepts are introduced into the language exclusively by either combining two or more kanji into a compound word, or as is more common, by taking an existing foreign term for that particular concept (in recent decades usually from English) and using the phonetic katakana script to approximate its sound.

2.1.1 Sets of kanji

Any research focussing on kanji usage or statistical analysis of kanji properties, regardless of the exact academic field, is by necessity limited to a specific subset of all possible kanji used in Japanese; the following subsets are commonly recognised as representative for the Japanese language, and are often used as predefined collections of "common" kanji in research. The aforementioned *Jōyō Kanji* set (p. 3) in particular provides a useful statistical baseline in terms of commonness.

In addition to the Jōyō Kanji, and its subset the *Kyōiku Kanji*, sometimes Japanese character sets — defined as Japanese Industrial Standards (JIS) — are used. The widely used JIS X 0208 standard consists of 6355 kanji, most of which[12] are used in the Japanese language, as well as Roman letters, punctuation and the two kana syllabaries. This standard is split up into two groups, referred to as

[11] Although in defence of our humble hypothetical sushi chef, he is more likely to be familiar with a number of kanji used exclusively for species of fish (e.g. 鰈or 鱒than our hypothetical professor, for whom these kanji might be fairly unfamiliar, not to mention quite irrelevant.

[12] After this standard was introduced, a small number of kanji in the standard turned out to be included by mistake. These so-called *ghost characters* (yūrei moji) have no known references in existing dictionaries and no recorded history of usage in Japanese, although some are real characters in other kanji cultures. Examples include: 𠮟 𡚴and 𣘺

level 1 (2965 kanji) and level 2 (the remaining 3390 kanji), where level 1 contains the kanji found to be more common (Nozaki & Ichikawa 1997, p. 25). Additional character sets exist, but they are rarely used in the context of psycholinguistic research[13].

In research published before the introduction of the Jōyō Kanji list in 1981, its predecessor the *Tōyō Kanji* (daily use kanji) list is sometimes used. This list was introduced in 1946 and consists of the 1850 most common kanji. Because the Jōyō Kanji list is essentially a superset of the Tōyō Kanji list[14] with an additional 95 kanji, useful conclusions can still be drawn from research conducted using the Tōyō Kanji set. An additional government-designated class of kanji that warrants mentioning is the *Jinmeiyō Kanji* (personal name use kanji) list. These kanji represent the set of characters officially permitted for use in newly registered names. The Jinmeiyō Kanji list consists of all Jōyō Kanji, supplemented with an additional 985 kanji[15].

2.2 Kanji readings

Before discussing the lexical processing of kanji, and its implications on the design presented later on, some statistical background on the way kanji can be read, may help explain the focus on sub-character features and kanji structure hereafter.

The Chinese characters used in Japanese can often be read (pronounced) in more than one way. Two categories of readings exist: *on*- and *kun*-readings. On-readings represent the original Chinese readings associated with the characters, adapted to the Japanese phonology. When kanji have more than one on-reading, these multiple readings often share a common historical ancestor. In the past,

[13] Most notably JIS X 0213 which extends JIS X 0208, and the international Unicode standard, but these consist of respectively over 10000, and well over 70000 Chinese characters. Consequently, they are ill-suited for use as a predefined set of common Japanese characters.

[14] With the exception of the kanji 燈(lamp, light), which was replaced by the simplified 灯in the Jōyō Kanji list.

[15] According to the Japanese Ministry of Justice's *Family register compatible character information* database: http://kosekimoji.moj.go.jp/kosekimojidb/mjko/PeopleTop (accessed 1st June 2009). 209 of these additional kanji are morphological variants of other kanji in the list — that is, kanji that differ only in shape with minimal differences, such as 亞(variant of 亜) and 祐 (variant of 祐).

three historical periods of significant cultural borrowing from China took place; the readings borrowed in those periods are, in order of chronology, classified as *go-*, *kan-*, or *tō*-on, referring to dynasties and regions representative of these three periods (Tamaoka 2005, p. 282). Because these periods were centuries as well as dynasties apart, the Chinese reading used on the Chinese mainland morphed throughout the years, causing different on-readings for a single kanji to often share a somewhat regular pattern in their readings. For instance, the go-on readings **myō**, **shō** and **kyō** usually correspond to the kan-on readings **mei**, **sei** and **kei**[16], respectively.

2.2.1 *Kun*-readings

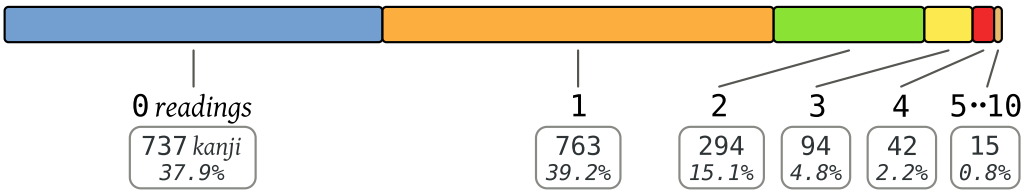


Illustration 2.1: Number of kun-readings for all 1945 kanji in the Jōyō Kanji list

A look at the number of kun-readings in the Jōyō Kanji list shows us the division as graphed in illustration 2.1 (statistics drawn from the database presented in Tamaoka & Makioka (2004)). Because the kun-readings existed as separate words in the Japanese vocabulary before the introduction of kanji, there is, unlike the on-readings, no direct correspondence between the kun phonology and kanji orthography at the sub-character level. However, kun-readings do repeat, sometimes occurring as the reading for a number of kanji; often the different kanji used for the same kun-reading signify different nuances of a similar meaning (Kess & Miyamoto 1999, p. 41).

While kun-readings may not be directly related to any sub-character components in kanji, they do play a significant role in the mental lexicon. Single kanji are likely to be pronounced by their kun-reading (Tamaoka 2005, p. 284), hinting

[16] Examples of kanji with these particular on-readings are, respectively, 名(name, fame), 青(blue), and 京(capital).

Base meaning	Word	Nuance
move/shift	移る	to move, to change, to drift
	遷る	to transition, to change
reflect	写る	to be photographed, to be projected
	映る	to be reflected, to be projected

Table 2.1: Kanji sharing the kun-reading *utsuru*

at possible primacy of kun-readings over on-readings. Although this claim is not without criticism (Kess & Miyamoto 1999, p. 39–40), kun-readings do appear useful within the mental lexicon, in part because of repeated readings, such as **utsuru** in table 2.1, interlinking several kanji.

2.2.2 *On*-readings

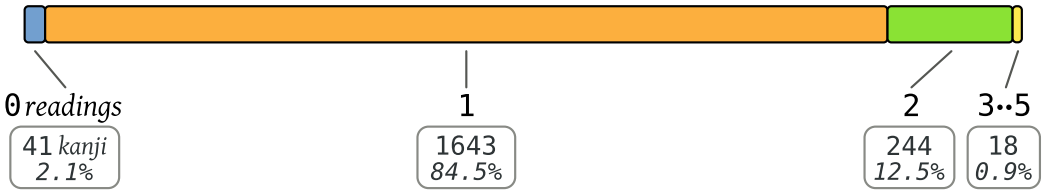


Illustration 2.2: Number of on-readings for all 1945 kanji in the Jōyō Kanji list

Arguably more engaging from a cognitive standpoint, are the on-readings. A look at illustration 2.2 (statistics drawn from the database presented in Tamaoka & Makioka (2004)) shows that the vast majority (97%) of Jōyō Kanji has either one or two on-readings (respectively 84.5% and 12.5%). Furthermore, where two on-readings occur, they are often phonetically related; in addition to the aforementioned historically related readings, other common relations are voiced and unvoiced consonant pairs (e.g., 存(exist, think) with **son** and **zon**), and vowel variations (e.g., 言(word, speech) with **gen** and **gon**).

2.2.3 Additional readings for Jōyō Kanji

It should be noted here that the kanji in the Jōyō Kanji list can have additional readings beyond those listed in the official chart, but such readings tend to be rarer in actual usage[17]. For a statistical analysis of the number of common on- and kun-readings, the Jōyō Kanji chart may be one of the few resources that can provide us with a fair overview of common readings, without cluttering the results with rare readings not in daily use.

2.3 Cognitive processing of kanji

Although computers treat kanji as the smallest indivisible unit of written text, humans do not process them in the same way as letters (Takebe 1989); just as the character 峠(*mountain ridge*, see illustration 1.4) can be split up into 山 上and 下, so can the majority of common kanji (around 90–93% of the Jōyō Kanji, and this percentage increases as more kanji are taken into consideration[18]) be split up into such distinct, distinguishable components. It is widely acknowledged that humans are aware of these sub-character features, and make use of them. One way of looking at this process is given by Tokuhiro (2003, p. 152), who uses a connectionist approach to describe the interconnected units encountered in the lexical process by means of a hierarchical model, where the process of reading a sentence containing kanji is shown to take place not only at the character, word, and sentence level, but also at the sub-character level, where the units are distinct, recognisable components found in kanji.

2.3.1 Kanji classification

Kanji are sometimes — often in older literature — described as being ideographic in nature, but only a handful actually are. Kanji can be classified in a number

[17] For example 約(*yaku*; approximately) has no kun-reading in the Jōyō Kanji chart, but it does in fact occur with a kun-reading as the (relatively rare) verbs 約まる (*tsuzumaru*; to compress) and 約める (*tsuzumeru*; to abridge), and the adjective 約やか (*tsuzumayaka*; modest).

[18] Based on a cursory analysis of the ideographic description sequence (IDS) data from the Kanji Database Project (Kawabata, accessed 16th July 2009).

of ways; the traditional six categories are the *rikusho* (六書: *pictographic*, *ideographic*, *compound ideographic*, *loan*, *semasio-phonetic*, and *extended meaning*.

Because this research focuses on the cognitive reality of kanji, this traditional classification system — although widely used — is only of limited usefulness to this research. The extended meaning category is mainly of historical relevance; kanji in this category have gained additional meanings over the course of history. For example, the kanji 長(*nagai*) originally only meant "long", but gained an additional meaning "chief/head", when used as a suffix in compound kanji words (e.g., 村長(*sonchō*, village chief)). Although interesting, this is not directly relevant to the cognitive processing of kanji, nor to the process of learning Japanese, whereas the other categories are — semasio-phonetic kanji in particular. Similarly, *loan kanji* originally had a different meaning, but were adapted to their current meaning at some point in history. Consequently, there is a tendency to leave out those two categories in psycholinguistic research, and focus on the remaining four (Kess & Miyamoto 1999, p. 35).

Another issue is the overlap that exists between the compound ideographic (characters formed by combining the abstract meanings of its two component characters), and semasio-phonetic (characters that combine a phonetic component with a semantic component) categories (Suzuki 2007, p. 62–64) if we follow a classification based on the *rikusho*, or — leaving out the loan and extended meaning categories — a division into pictographic, ideographic, compound ideographic and semasio-phonetic kanji. This distorts the actual number of kanji that exhibit the characteristics of semasio-phonetic kanji. In fact, an overwhelming majority of kanji can be considered semasio-phonetic; Suzuki (2007, p. 64, drawing statistics from the *Kanji Hyakka Daijiten* (Satō, 1996)) concludes that 1012 (52%) of the Jōyō Kanji are "pure" semasio-phonetic, with an additional 514 (26.3%) falling in a mixed compound ideographic/semasio-phonetic category. Thus, following Suzuki (2007), a total of 1526 (78.5%) of the Jōyō Kanji can be considered (functionally) semasio-phonetic. This matches earlier estimates, such as Itō (1979, p. 69), who estimated that 80 to 90% of all kanji used in Japanese are semasio-phonetic, and Kess & Miyamoto (1999, p. 35), who place the number nearer to 80%. The fact remains that such a large majority warrants attention to the way they are treated in the mental lexicon.

Incidentally, the less common a set of kanji is, the more semasio-phonetic kanji appear in it; a look at the six grades of the Kyōiku Kanji chart, reveals that there is a strong concentration of ideographic and pictographic kanji in the sets of kanji that are taught in the lower grades (Tamaoka & Makioka 2004). From a pedagogical standpoint this makes sense, because these characters are often simpler in shape and composition, and tend to form the basis for more complex characters.

2.3.2 Semasio-phonetic kanji

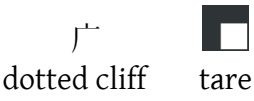
Kanji	Radical	Position	Meaning	Sample usage
底	广 dotted cliff	 tare	bottom	底辺 **tei**-hen (*base*)
低	亻/ 人 man	 hen	low	最低 sai-**tei** (*the lowest*)
抵	扌/ 手 hand	 hen	resist	抵抗 **tei**-kō (*resistance*)

Table 2.2: Semasio-phonetic kanji sharing the phonetic component 氐(**tei**)

The basic type of semasio-phonetic kanji is composed of a semantic, and a phonetic component. In most cases, this semantic component corresponds to the traditional radical. Table 2.2 shows three kanji that share a phonetic component read as **tei** (氐[9]), but have differing radicals. Semasio-phonetic kanji appear in the form of a number of spatial configurations, the most common of which is the *hen* plus *tsukuri* composition; a kanji with its radical and phonetic

component side-by-side. Within JIS X 0208 level 1 (the 2965 most common kanji), as much as 1668 kanji can be regarded as such *left-right* separable characters, composed of a total of 857 unique parts (Saito et al. 1995*a*, p. 117).

Direct	Indirect			None
				
kaku	**kyaku**	**raku**	**raku**	**ro**

Table 2.3: Semasio-phonetic kanji with direct, indirect and unrelated on-readings, sharing the phonetic component 各(**kaku**)

The relation between phonetic component and actual reading is not always one-on-one. Readings can vary slightly from the common reading of the phonetic component itself (see table 2.3). In Saito et al. (1995*b*) the characters that share the same reading as their phonetic component are called *direct accessed characters* (such as 閣which is read **kaku** just as its phonetic component 各is), and the kanji that are read with a phonetically related variant of the phonetic component's reading are called *indirect accessed characters* (such as 客, 洛and 落 in table 2.3). Notice that 落(*falling*) is formed by taking (the fairly uncommon) 洛(*capital, Kyoto*) as its phonetic component, and adding the grass radical 艹; kanji can occur as part of another, and it is not uncommon for the phonetic information to be inherited by the bigger kanji. Phonetic components are not necessarily kanji in use in modern Japanese[20]; readings accessed through phonetic components such as 灸 (**ken**) as seen in 験and 検often do not have a meaning as a stand-alone character (Saito et al. 1995*b*, p. 92).

[19] 氐(**tei**) does have a meaning — it represents the "Di" people, an ethnic group that lived in China until the mid 6th century — but it is not in common use as a separate kanji. Many phonetic components are actually Chinese characters in their own right — although often obscure — even though they are never used in common Japanese.

[20] Although with the ongoing expansion of the Chinese character sections of the Unicode standard, a lot of these components can already be used on modern OSs as proper characters, provided a font that includes them is installed or present in the document. This document, for

Any learner of Japanese will eventually — to some extent — notice this link between phonetic components and recurring readings; he will most frequently encounter the *left-right* kanji, where the right part is most often responsible for allowing the retrieval of phonetic information (Saito et al. 1995*b*, p. 90) and this characteristic is the most obvious, but this facilitating effect is not limited to this specific spatial arrangement (Tamaoka & Yamada 2000, pp. 206–207). Itō (1979, p. 75) hypothesises that knowledge of phonetic and semantic elements of kanji can aid the learner in the process of acquiring a kanji vocabulary; actively comparing kanji with the same phonetic component (such as the three kanji in table 2.2), but different semantic components would help to further deepen the learner's understanding of Japanese. As there is a strong interdependency between kanji phonology and kanji orthography (Tamaoka & Yamada 2000, pp. 204–206), could visualising these relations perhaps aid the learner?

2.3.3 Multiple paths to kanji cognition

Semantic processing of kanji does not necessarily go by way of the phonetic properties of the characters. Instead, it is commonly held that this process takes place mostly at the morphological level. This is in contrast with alphabet-based languages, where meaning relies heavily on the phonetic information represented by the sequences of letters (Kess & Miyamoto 1999, p. 45). Saito et al. (1995*a*, p. 114) for instance, observed that while homophony between kanji is a factor in inducing false recognition of the character's meaning, this tends to occur only when the kanji also share an identical component. Clinical trials confirm that phonological information in kanji significantly influences the lexicosemantic process (Wei et al. 2007).

The process of activating the mental lexicon when processing kanji should not be seen as a binary choice, with activation going via either orthographic or phonetic routes. More likely, depending on the kanji being read (and the person reading it) there exists a great deal of interaction between both pathways, and even some degree of concurrency (Kess & Miyamoto 1999, p. 197).

example, includes the font necessary to display characters such as 貪, by virtue of the GlyphWiki project (see section 4.1.1).

2.3.4 The role of radicals as semantic classifiers

In theory, radicals may act as semantic classifiers, but as Kess & Miyamoto (1999, pp. 67, 201) point out, the traditional semantic meanings attributed to radicals should not be taken as *inevitable* first steps in the process of deciphering a character's meaning. These radicals can offer vague hints to the meaning of the kanji, but the lexical process does not appear to follow a set sequence of *first* resolving the global semantic grouping and *then* the exact meaning. However, these semantic components do appear to contribute to the whole process of deciphering kanji *in parallel* with the other character properties, especially when the kanji is unfamiliar (Kess & Miyamoto 1999, p. 68).

To more precisely determine the function of radicals Hirose (1999) conducted an experiment using a selection of semasio-phonetic kanji with a clear-cut *hen* plus *tsukuri* structure (such as 低and 抵in table 2.2). In this experiment Hirose (1999) too concludes that the semantic radical certainly contributes to the cognitive process, but its effect is dwarfed by the influence of the phonetic component. Tamaoka & Yamada (2000, p. 205) concluded that there is a strong and significant correlation between knowledge of kanji phonology, orthography and semantics, further suggesting that there is a significant degree of interconnectedness between the components that represent these kanji characteristics and the way they are processed.

Taking the above into account, we might conclude that there is reason to deemphasise the traditional focus on just the one "correct" radical, or — similar to the universal radical index in *The New Nelson* (Nelson & Haig 1997) — only all of the components that also happen to be radicals from the list of 214 traditional radicals. Instead, for a modern electronic kanji dictionary it might be preferable to let the user explore and use *all* recognisable components of a character, even when those elements are not considered proper radicals. Ideally, a graphical user interface would allow the user to easily identify the properties of any com-

[20] Surprisingly, the study presented in Tamaoka & Yamada (2000) also found that knowledge of the correct stroke order does not appear to be linked as closely as phonetic, semantic and orthographic information in the kanji lexicon, although they do conclude that stroke order has an effect on knowledge of kanji components.

ponent of kanji, regardless of whether it is a traditional radical, a kanji in its own right, both, or neither.

This does not mean that knowledge of what part of a character is its radical is not useful; after all, traditional dictionaries and reference works remain a valuable source of information, and existing electronic dictionary software (including PEDs the user may already have in his possession) often do rely on radical and stroke knowledge. Nor is it necessary to leave out the traditional radical and stroke count method from the designs proposed in chapter 3; the idea is to provide multiple ways to construct queries — just as the cognitive processing of kanji can follow different, parallel paths — and let the user decide upon the method that best suits his needs.

2.3.5 Kanji components

Humans store graphical information in a structured manner, not as static images. Kanji too, get processed in this way; a meaningful interpretation of the visual data is formed by processing the kanji in chunks (Nozaki & Ichikawa 1997, p. 26). In other words, we are aware of a complex character's sub-character features; its components. Moreover, information about kanji sharing the same radicals or components is represented in the mental lexicon as meta-knowledge, and plays a facilitating role in character recognition (Saito et al. 2002, p. 507). Components other than just the radical often carry semantic properties as well; many are kanji in their own right, or variant shapes of radicals and kanji — such as 氷(the bottom-right component in 緑(*green*)) being a variant form of 水 (*water*). Would awareness of the meanings of these components facilitate our understanding of the character?

Remembering kanji through mnemonics

The beneficial effect of kanji component knowledge is not limited to access to phonological information, rough semantic information and cognitive priming through its radical. One method of remembering the meaning of such kanji, is to actively use the meanings of its constituent components by creating mnemonic sequences, thus forming an aide-mémoire. Takebe (1989) presents a compen-

dium of such sequences in his work for all of the kanji in the Jōyō Kanji list; table 2.4 lists a small number of these (translated) sequences. Other examples of pedagogical works employing mnemonics include Heisig's (2007) popular *Remembering the Kanji* series, which teaches the meaning of kanji through elaborate mnemonic stories. While similar to Takebe's shorter sequences, the meanings contributed to the various components are made up rather than based on their actual meanings.

Kanji	Meaning	Mnemonic
引	*pull*	弓is the kanji for *bow* and the vertical bar could be interpreted as an *arrow*; to fire an arrow, you **pull** back the bow
宿	*lodge*, *dwell*	宀is a *roof* that provides **lodging** to a lot (百*hundred*) of people (亻 is a variant of 人*people*)
苗	*seedling*	艹is the *grass* radical, and 田is a *cultivated field*; **seedlings** are the "grasses" that grow in cultivated fields
猫	*cat*	苗is the kanji for *seedling*, seedlings are flexible; 犭is the *animal* radical; **cats** are flexible animals

Table 2.4: Samples of mnemonic sequences for kanji

Still, on its own, the effectiveness of this method is debatable if it is not actively combined with teaching pronunciation and usage of the kanji. However, the popularity of *Remembering the Kanji* in particular, does provide some indication of the validity to the call for stimulating a more active understanding of kanji components[21]. While Nozaki & Ichikawa (1997, p. 26, 28) argue that knowledge of the components used in complex kanji aids in the learning of new characters, concluding that there is a significant correlation between the acquisition of component discernment skills, and the learning of new kanji. Tokuhiro (2003, p. 153) too stresses the importance of teaching L2 learners at least basic knowledge of kanji elements and radicals.

[21] The mnemonics in Heisig's (2007) work may not use the actual meanings of the components used, it does teach the learner to more actively recognise and utilise kanji components.

2.4 Facilitating the learning process

To successfully learn a second language and its script, it needs to be seen used in its proper context; L2 learners need multiple exposures (Chikamatsu 2005, p. 89), in various usages, to words and kanji in order to understand their correct usage, meaning and nuance (Mori 2003, p. 410). The same holds true for the sub-character features outlined above; without context, teaching about kanji sub-character characteristics may even be harmful, rather than helpful (Chikamatsu 2005, p. 90). How can electronic dictionary software aid the learner of Japanese in the learning process? Mori (2003, p. 414) suggests actively stimulating L2 learners to seek out contextual information; my hypothesis is that the GUI of an electronic dictionary software application is a very intuitive place to provide this context. After all, using a dictionary often implies that an unfamiliar or poorly understood word or character is encountered in some text being read or composed — the usage context is provided by whatever is being read by the learner at that moment, or by if he is writing something of his own composition, by example sentences provided and usage hints for that entry in the dictionary.

Knowledge of radicals and kanji components correlates with knowledge of kanji phonology, orthography and semantics (Tamaoka & Yamada 2000, p. 205), so the focus lies on providing easy access to this information, as well as allowing the user to leverage it. An experimental kanji learning application proposed and implemented by Nozaki & Ichikawa (1997, p. 29, 32–33) proved successful in teaching students insight into the structure of kanji and improving their kanji retaining abilities. This application explicitly visualised the position of the radical of complex kanji (similar to the information shown in table 2.2), thus providing the user with visual feedback aiding the subconscious discernment of sub-character components.

In essence, the design presented hereafter should provide a way to visualise the spatial layout of complex kanji, and providing the user with information on the function of components. Obviously, this includes pointing out the traditional radical and its meaning, but also which component — if present — carries the phonetic information, and the meaning of other components. In addition to providing information on existing kanji, this sub-character approach

should also be able to help in the act of searching for kanji. Given the above, it is plausible to assume that the substitution of kanji components is beneficial to L2 learners (Takebe 1989, p. 83–85). In Hoek (2007, p. 18–23) the notion of sub-character search queries was briefly entertained, but the conceptual design presented therein seemed to complex for the average user. One reason is that this design would have required the user to construct the query from the ground up; beginning from scratch, the user would have had to select spatial configurations and add components (selected as radicals, or as kanji through an IME) to those configurations. However, there are many cases where the kanji (or word) sought after contains features familiar to the user, such as another kanji he does know, or components from such kanji.

Consider the kanji 綿(*cotton*), which is part of the Jōyō Kanji chart, but not as common as 線(*line*[22]). A learner encountering the former for the first time may very well be aware of its composition up to some extent; he may notice, for example, that the 糸(*thread*) radical takes up all vertical space on the left, and the right consists of two components, stacked vertically: the kanji 白(*white*) and, …something else. Nevertheless, he does notice that this *something else* bit of the kanji is the only part that distinguishes it from 線a kanji he does know, and for which he knows the reading too — entering this kanji in a digital environment requires little effort with the IME. For this user it may very well be faster, and more intuitive, to simply enter the kanji 線select the bottom-right part and delete it, and ask the computer to return a list of all kanji that match his newly constructed query. If the dictionary software orders the results list by commonness (a sensible default), the top two results will be 線and the desired 綿– since these are the only characters in the Jōyō Kanji list that apply[23].

A different example using multiple kanji might be the (hypothetical) case of a more intermediate learner perplexed by the compound kanji word 躊躇(**chūcho**, *hesitation*) encountered in a more complex text he finds himself reading. These kanji are not part of the Jōyō Kanji chart, and in general only occur

[22] 線is also used as a suffix for railway-lines, and thus quite common in the urban Japanese streetscape. Anyone who has ever been to Japan with even a modicum of kanji knowledge will have noticed this kanji at some point.

[23] Based on an analysis of the ideographic description sequence (IDS) data from the CHISE project (section 4.1.2).

as this specific pairing. The kanji may be complex, but this user would not be an intermediate learner if he did not at least recognise the kanji (and radical) 足(*leg*) positioned on the left of both kanji. Rather than counting strokes and searching for one of these characters, and *then* searching for the compound word, why not let him take a shortcut and search for any two-kanji compound words where both characters have the 足radical on the left?

These hypothetical[24]examples not only provide the user with more ways to accomplish his goals — that is, to find kanji or kanji compound words in the dictionary — they also act on the cognitive level, by providing the user with positive feedback on his understanding of kanji composition and capability of component discernment.

2.5 Second language acquisition: nature versus nurture

Although this thesis focusses on designing dictionary software for L2 learners of Japanese, some of the research on kanji cognition cited in this chapter reports on experiments where exclusively native Japanese speakers were used. The rationalisation for this is the limited availability of research focusing exclusively on cognition of kanji with L2 learners from a non-kanji cultural background. Can such research focused on Japanese L1 subjects be safely used in the context of L2 learners?

Despite the differences between the processes of native language acquisition and learning a second language, it is arguable that these discrepancies do not invalidate the use of such research as the foundation for a software design aimed primarily at L2 learners. The biggest issue at hand here is the theoretical dichotomy of language *acquisition* versus language *learning*. While native speakers of Japanese enjoy the benefit of acquiring the skills to speak, read and write Japanese in a natural environment — that is, an environment where the language is heard and seen at all times — before reaching adulthood, most L2

[24] Although they are, for the most part, based on the author's personal experience (and occasionally frustration) as an L2 learner of Japanese.

learners of Japanese are already adults when they start, and lack this beneficial immersion in the culture where the language is naturally used. One traditional view on second language acquisition holds that this dichotomy is what separates native and L2 learners. Takebe (1989, p. 161) for instance argues that Japanese children acquire an innate capability to discern the semantic, phonetic and other properties of kanji unconsciously, whereas for L2 learners this is a much more conscious process.

However, regardless of whether or not this acquisition-learning hypothesis is valid or not when looking at the process of learning a foreign language as a *whole* — including grammar rules, proper use of honorifics and vocabulary — the process of learning the *kanji* is not exclusively a matter of conscious learning. Tokuhiro (2003, p. 169) too concludes that for L2 learners of Japanese this process has both the obvious conscious facets expected from a L2 learner, as well as subconscious facets. Chikamatsu (2005, p. 87, 89–90) argues that the biggest weakness for L2 learners is that the cognitive network that links sub-character components and spatial composition awareness with phonology and morphology, is lacking in comparison with native users of Japanese[25].

[25] Chikamatsu (2005) bases this on experiments with L1 and L2 learners, where the tip-of-the-pen phenomenon is used to measure what information the subjects know about kanji that they feel they know, but at that moment cannot reproduce. The results showed that where native users could often at least identify the correct radical and the overall shape of the target kanji, L2 learners made significantly more mistakes in this area.

Chapter 3

Interface design

The works cited above help to illustrate that while L2 learners are able to acquire a kanji lexicon capable of picking up on sub-character features, experiments analysing L1 and L2 errors in writing kanji show that the L2 lexicon tends to be more limited in nature, especially for beginners and intermediate learners. To assist this group, explicitly teaching them about sub-character features (Chikamatsu 2005, p. 89), and allowing the use and manipulation of components and their spatial arrangement, are two possible approaches. The former may be achieved by visualising this information, similar to the way this is done with the illustrations in chapter 2 (tables 2.2 and 2.3). However, for this to be successful, this information has to be presented within its proper context. Adding this functionality to the user's electronic dictionary seems a logical addition to the basic dictionary functionality already provided, with the added benefit of providing context through the user's searches. In essence, providing information on phonetic components, and the meanings of all distinct components that have one, can be seen as an extension of the duties of a kanji dictionary. The other approach — allowing users to construct search queries using sub-character features — can be used for both kanji and word dictionaries. How can the conventional approach to electronic kanji and word dictionaries for the Japanese language be augmented to allow for these two concepts?

Following the requirements specifications process as outlined in Mayhew (2008), the sections in this chapter represent the first steps in the design process

for a software application capable of visualising sub-character characteristics, and aiding the learner in acquiring or extending his kanji vocabulary, as well as allowing for the construction of search queries such features.

3.1 Requirements analysis

First, an analysis of the user's needs is in order, enumerating both the basic tasks which the application should enable the user to perform, as well as the desired additional functionality focussing on the sub-character properties of kanji.

3.1.1 User profile

In essence, any user of Japanese in need of an electronic dictionary — including native Japanese — could benefit from the software proposed in this thesis, but the main target audience consists of L2 learners of Japanese with a non-kanji cultural background, as this group stands to benefit the most. In terms of Japanese language proficiency, there is essentially no upper bound, but a minimum of practical experience with the Japanese script — kanji in particular — can be seen as a fair prerequisite. Some familiarity with traditional dictionaries, and the more common radicals in the classical radical chart, can arguably be expected from any learner of Japanese after a few months of study, as well as knowledge of how to operate an IME for the Japanese language.

Because Japanese as a second language is taught worldwide, the target group is extremely heterogeneous in nature. Users hail from a wide variety of cultural backgrounds, and their preferred language[26] for using software can be anything. Consequently, the dictionary resources they require are not limited to just a Japanese–English bilingual dictionary and a kanji dictionary with English meanings. The degree of computer literacy too will vary greatly, as well as the preferred setting in which the user wants to use the software; or more concretely, the computing environment can be any platform upon which the user can install

[26] Often, this will be the user's native language, but not necessarily. Personal observations suggest that users with a higher degree of computer literacy may often prefer English over their native language, especially if they deal with English resources on a daily basis.

custom software, with OS and device class (ranging from desktop computer to netbook and smartphone) being major variables.

Usage patterns and frequency can be expected to be similar to that of paper dictionaries and their electronic counterparts. In other words, usage is likely to be highly irregular, with periods of intensive usage (reading Japanese articles, working on Japanese texts, etcetera) alternated by periods of only the occasional look-up.

3.1.2 Task analysis

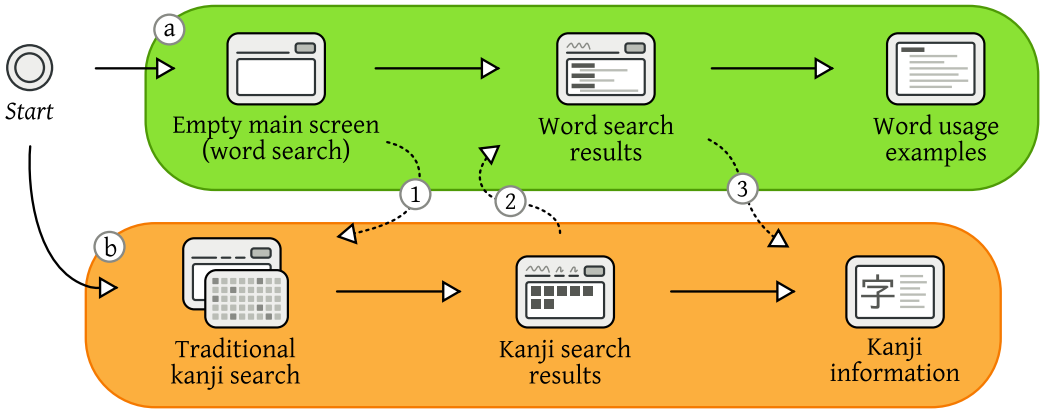


Illustration 3.1: Workflow illustrating word (a) and kanji look-up (b) using conventional electronic dictionary software

The primary task a user will perform with a dictionary, is looking up unknown, or unfamiliar words and characters. Currently, looking up words or characters using electronic dictionary software follows a (simplified) workflow resembling the illustration 3.1. A basic word search is conducted by entering the word in Japanese using an IME to convert the input to kanji, or — if the correct kanji are not known to the user — by entering only its reading. Alternatively, a second language may be used, provided bilingual dictionary resources supporting that language are available[27]. The search result list shows

[27] Fortunately, a number of high quality, free software, bilingual Japanese dictionaries exist. Well-known examples include Jim Breen's Japanese–English *EDICT* (`http://wwwjdic.com`) and Ulrich Apel's Japanese–German *Wadoku Jiten* (`http://wadoku.de`).

the basic glosses for the entries that match, including word class, usage hints (possibly including verb conjugations, word commonness, and tags indicating profane, colloquial, archaic, or slang entries), etcetera. Optionally, the glosses can be linked to example sentences, providing usage context. Kanji searches can be performed by using the traditional radical and stroke count techniques, often augmented with methods to directly enter kanji by drawing it (as described on p. 11) or simply entering the desired character by means of the IME. Most dictionaries provide kanji search functionality as an auxiliary option available from the main dictionary screen (path ① in illustration 3.1), adding a step to the workflow. A search for words containing the kanji found through the kanji search, can sometimes be directly accessed by clicking on the characters shown in the kanji search result list (path ②); when this is not possible, users may copy and paste the relevant character from the kanji search list into the input field for a new word search. Either way, this appears to be a fairly common operation — necessary when looking up an unknown multiple kanji compound word where the user is unable to use the IME to enter the kanji — that is currently somewhat inefficient. Information on specific kanji used in words shown in the word search result list can usually be directly referenced (path ③).

In general, PEDs, and dictionary software running on netbooks, PDAs, and smartphones, are used as portable dictionaries. The text the user is concerned with, will often be a physical resource such as a book or a printed article. In computing environments where more screen real estate is available (desktop computers and laptops), the dictionary is likely to be used next to a digital document open on the desktop as well. From the user's perspective, both use cases necessitate that the dictionary software can be used without taking up large amounts of screen real estate.

3.1.3 Usability goals

From the user profile and task analysis above a number of practical usability goals can be distilled:

Avoid clutter Regardless of the potential benefits of bringing the sub-character characteristics of Chinese characters to life, the user's primary goal in

using a dictionary is to look up unknown, or unfamiliar words and characters. An important goal to keep in mind, is that additional information about the kanji and its components should be readily accessible, but it should not clutter the interface when the user simply wants to look up a word. Similarly, browsing for any additional information on kanji should be possible without taking the user away from his current search results.

Window size When a desktop allowing multiple concurrent windows to be present is being used — that is, the display resolutions can accommodate them, and the windowing system used allows for this — the application as a whole should be small enough to not get in the way of what the user is doing with other applications.

Localisation Given the diverse nature of the expected user-base. In the long term, localisation — especially translation — of the application should be an option.

Additionally, from the treatise on the cognitive reality of kanji in chapter 2, the following concrete goals can be set:

Identify kanji components Aid the user in acquiring kanji by visually dissecting the various components of a character, and showing their semantic and — if relevant to the current character — phonetic properties.

Interact with kanji components Allow the user to construct kanji and word search queries using not only kanji or kanji readings, but also sub-character features: spatial layout and components.

3.1.4 Platform capabilities and constraints

Modern OSs are already capable of using and displaying a large amount of Chinese characters. Microsoft's Windows Vista reportedly has 70000 Chinese characters available (Nomura 2007, p. 140), and the GNU/Linux OS supports any character as long as it is defined in the Unicode standard, and a font capable of displaying it is installed. The same holds true for the Japanese IME; inputting Japanese text is possible on all major desktop OSs and most of the portable OSs.

One extreme variable across the various platforms is display size. Since the introduction of the desktop computer, the physical size of the display portion of the monitor increased steadily, and the amount of pixels it is capable of displaying has skyrocketed[28]. Yet at the same time, on the lower end of the spectrum, the current generations of smartphones — in accordance to the limitations of their physical size — sport much lower display resolutions[29].

3.1.5 Design guidelines

In addition to the usability goals listed above, a number of general design guidelines should be taken into consideration. The following list is not definitive; should user feedback, or faults detected during the implementation phase provide additional hints with regard to the general design, this list will be augmented accordingly.

Windowing When every Chinese character in the results list can be clicked for additional tasks, it becomes easy to lose track of the task initially set out to perform. In order to allow the user to freely browse for any additional information he desires, without giving up on the search results for the task at hand, the tabbed-browsing[30] model seen in modern web browsers can be used. In effect, browsing for additional information on kanji, or looking up usage examples for words, resembles the way we use the world-wide web — imagine looking up the meaning of some foreign concept on Wikipedia or Google (our main task) and getting slightly sidetracked by opening hyperlinks to related, interesting topics in new browser-tabs, whilst leaving the search result for the main task open in another tab for later perusal. The advantage of this approach, is that it can be used in situations with limited screen real estate, as well as on larger desktops, where detachable

[28] display resolutions of 1920×1080 pixels are common for the current generation of flat-panel monitors, and even laptop computers often come equipped with 1440×900 pixel displays.

[29] A 'mere' 480×320 pixels presently forms the lower bound for these devices — this is the display resolution of the current generation of popular smartphones, such as Apple's *iPhone* and HTC's *Dream*.

[30] Modern web browsers and many other applications make it possible to open a new instance of that application in a so-called *tab*. Using the office metaphor of tabbed pages in a binder, clicking on a tab shows the corresponding instance.

tabs can be dragged out of the current window to become a new window on its own. Whether information is opened in tabs or not, can be influenced by the user — either through the application's configuration, or by holding a modifier key[31].

Performing actions Intuitive drag and drop, menus, and keyboard shortcuts should all be utilised to allow the user to perform actions through multiple means. In general, all actions that can be performed with mouse should have keyboard counterparts (Benson et al. 2008, § 10.1.1). Where possible, direct manipulation of the objects the user is working with should be possible, rather than explicitly requesting actions via menus (Benson et al. 2008, § 1.9).

Colour Although the design presented here makes ample use of colour as a visual cue, care is taken to limit the function of colouring to providing *additional* information. While tempting to exclusively depend on varying hues to make different types of interface elements distinguishable, colour should ideally only be used to provide redundant information. This is because an estimated one in ten people suffer from some form of colour-deficiency, as noted in Watzman & Re (2008, p. 348) and Benson et al. (2008, § 8.1). The colour palette used in these mock-up figures is the Tango colour palette[32]. Consistently using a predefined colour palette composed of colours that match well with each other (Watzman & Re 2008, p. 348) is beneficial to the user experience, and has the additional benefit — if the technological implementation allows for it — of being replaceable; that way, any user can define his own "colour theme" as well, if need be.

[31] For many platforms and browsers the control modifier is used for this function.

[32] Available at `http://tango.freedesktop.org/Tango_Icon_Theme_Guidelines`.

3.2 Conceptual user interface design

3.2.1 Conceptual model design and mock-ups

Because the following design is conceptual, practical and common GUI elements such as the menu bar and window title are omitted, instead the focus lies on the novel components of the proposed interface. As this design extends the existing basic electronic dictionary GUI rather than reinvent the wheel, the standard word search design does not change significantly and will likely end up similar to illustration 1.5. One important feature however, is that any kanji in the word search list can be clicked in order to access its entry in the kanji dictionary.

Kanji information

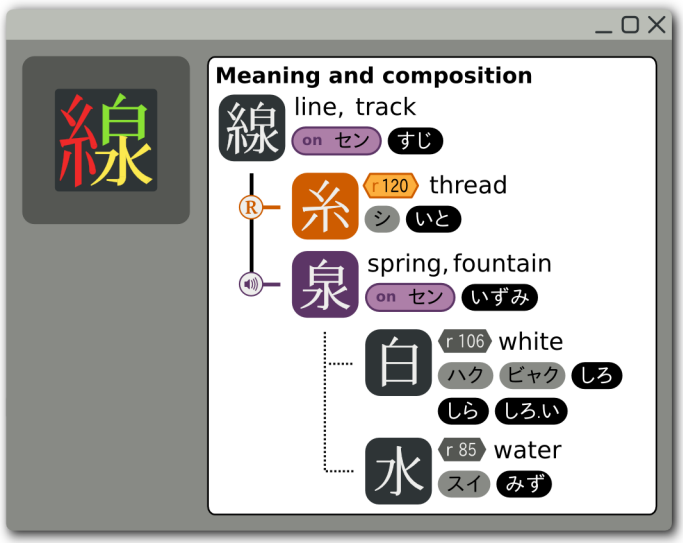


Illustration 3.2: Kanji information pane showing composition and readings for the kanji 線

The kanji information provides the user with the means to easily access information on the sub-character features of kanji. Illustration 3.2 shows the spatial composition for the kanji 線(*line*, *track*), indicating radical number (if the component is also a traditional radical), meaning and readings for its components. For the sake of clarity, kanji name-readings and other additional informa-

tion have been omitted from the component entries. 線borrows its on-reading (セン, **sen**) from the component 泉(*spring*, *fountain*); this is accordingly visualised in the illustration, as is the component that is traditionally recognised as its radical (the radical 糸). In this example the readings are shown in katakana (on) and hiragana (kun), which is the Japanese way of writing readings, but for a beginner it may be more convenient to use the Hepburn romanisation system; the application should allow the user to configure his preferences accordingly (Hoek 2007, p. 17–18). The same goes for the kanji dictionary resource used; in this example, the English meanings were drawn from Jim Breen's KANJIDIC[33], but if a similar resource in another language is available, that could be used as well. Illustration 3.3 shows what happens when the mouse cursor moves over one component entry in the list, thereby highlighting that row; the corresponding section of the colourised kanji lights up as well — or rather, the other components are toned down. Its function is to indicate the location of a component within the kanji and provide positive feedback to the user, further reinforcing the link between kanji and its components.

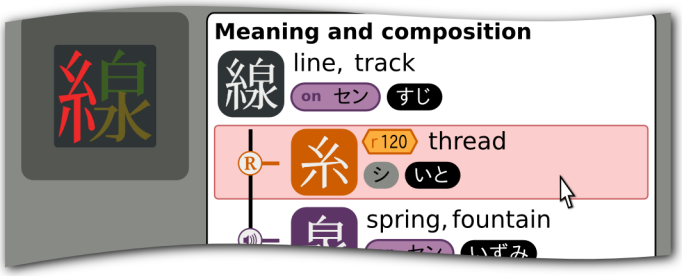


Illustration 3.3: Fragment of the kanji information pane highlighting the 糸component

The Kanji Navigator

The goal of interaction with kanji components may be satisfied with the following concept — tentatively named *Kanji Navigator*. This is an alternative to the traditional kanji search methods described earlier (p. 5). Note that these traditional methods will still be available to the user; the *Kanji Navigator* is presented as an

[33] A kanji dictionary companion to the bilingual Japanese–English EDICT dictionary, available at `http://www.jdic.com`.

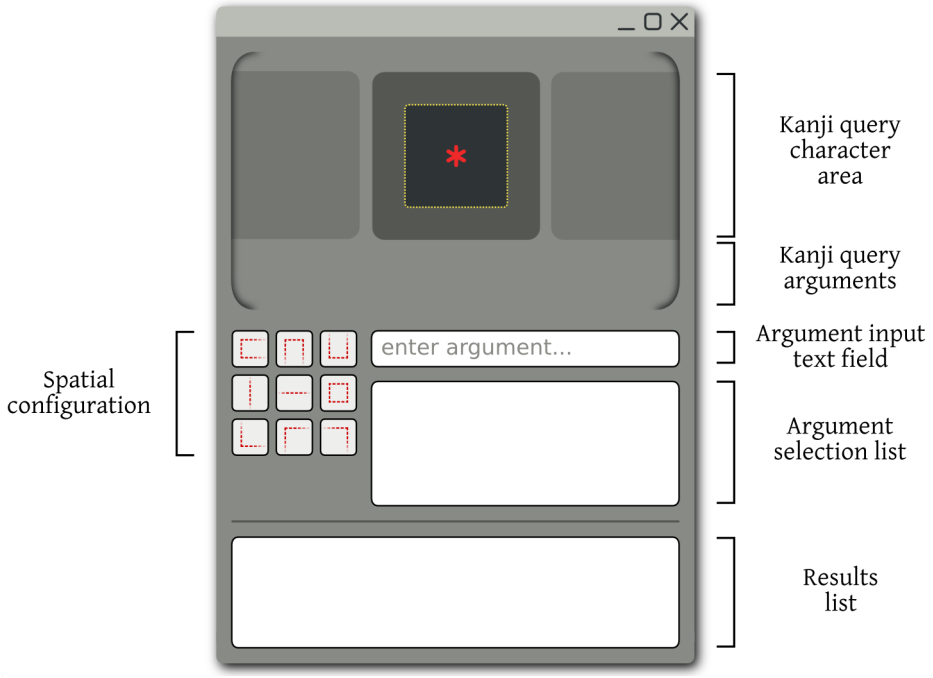


Illustration 3.4: The Kanji Navigator in its initial state

additional means of achieving the user's goal. The initial state of this component is shown in illustration 3.4.

A practical example may help explain its usage; one of the troublesome cases mentioned above (p. 28) is 躊躇(*hesitation*) — assuming the user cannot simply copy and paste this word from a digital resource. The user is not (yet) able to recognise the word or its two kanji, but he does know that both kanji have the component 足(*foot*) on the left side. In illustration 3.5 the steps our user has to take to enter his query are shown. By default, an empty placeholder is shown with the asterisk representing "anything". First the user splits ⓐ up this placeholder into two parts ⓑ by selecting one of the nine spatial configuration buttons. Already, the *Kanji Navigator* will start to list all kanji that match — all kanji with a top-level vertical split. Of course, the resulting list is still unwieldy, so he adds an argument to his query by using his IME to enter the kanji 足ⓒ. This character can be represented by a variant form as well — 𧾷to be exact — but because this is only a slight variation of the same basic shape, the *Kanji Navigator* initially offers an argument that comprises both variant forms of 足

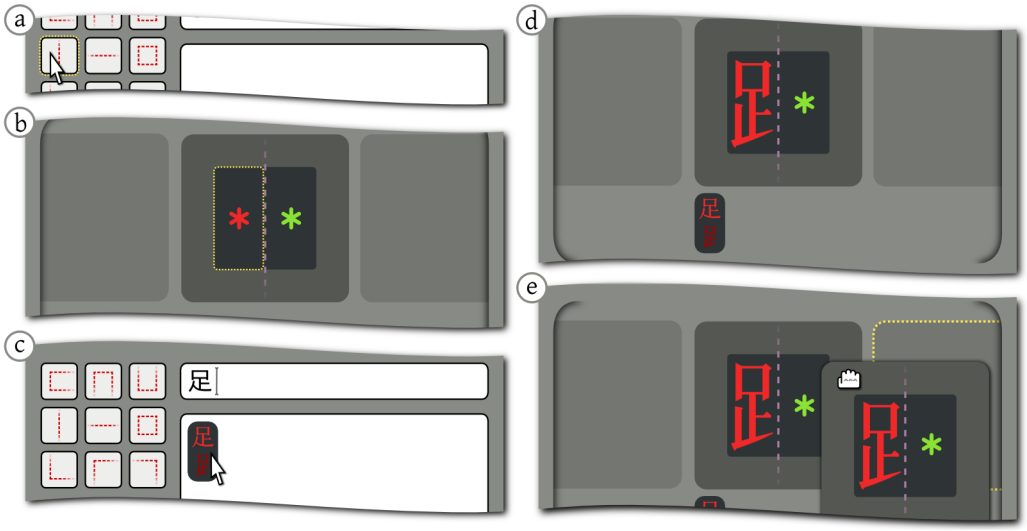


Illustration 3.5: The steps needed to enter a query for any two character compound with 足on the left of both kanji

By dragging the 足argument into the left part of the kanji query ⓓ, the *Kanji Navigator* further filters its results list to now only include kanji with 足on the left, and anything else on the right. To extend this query from kanji search to word search, the user needs to add the same argument and spatial arrangement to a second kanji. One possibility would be to click on one of the empty squares next to the current kanji query, and repeat steps ⓐ to ⓓ, but because it is the exact same argument, simply dragging and dropping[34] the existing one into the empty square next to it will suffice ⓔ — not to mention being much faster. Either way, the results list most likely will list 躊躇as the only possible option.

The arguments that can be used in the query are not necessarily limited to character components. In this 躊躇example, had the user been familiar with the on-reading commonly associated with the component 著(the reading **cho**), he could have used that as an argument as well; if the user input in the text entry widget is a valid on- or kun-reading, the *reading* argument becomes available in the selection area (illustration 3.6). Similar to the component argument it can be

[34] More concretely, following common desktop environment guidelines, simply dragging and dropping the argument should simply move it to the next square, whilst the same action accompanied by a modifier-key such as control can be used for the "copy" action (Benson et al. 2008, § 10.1.3).

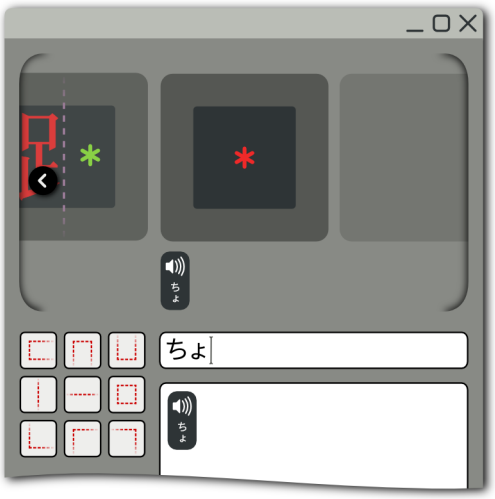


Illustration 3.6: Using a phonetic argument to search for 躊躇

dragged to the kanji query area, but unlike the component argument it cannot be assigned to a specific part of the kanji; the scope of the argument is the whole kanji.

Starting from scratch is not a necessity; in fact, often using an existing kanji as the basis for a new query may be faster. The earlier example of 線(*line*, *track*) and 綿(*cotton*) is such a case — illustration 3.7 shows the steps the user might take. Note that entering 線in the kanji argument input field ⓐ makes the widget list not only the kanji itself, but also all of its components. Although a lot of kanji components are simply kanji or radicals in their own right, there is a large group of components that are not commonly used as such. In these cases, it seems fruitful to allow the user to use his present knowledge of kanji and their components; if he knows that the component he is looking for is part of a kanji he knows, then allowing him to enter it as an argument via that kanji is certainly faster than using some convoluted component selection procedure. In this example the user finds the character he was looking for by dragging the whole character 線into the kanji query character area ⓑ and removing the 水(*water*) component by using the context menu ⓒ (accessible through a right mouse-click in most desktop environments).

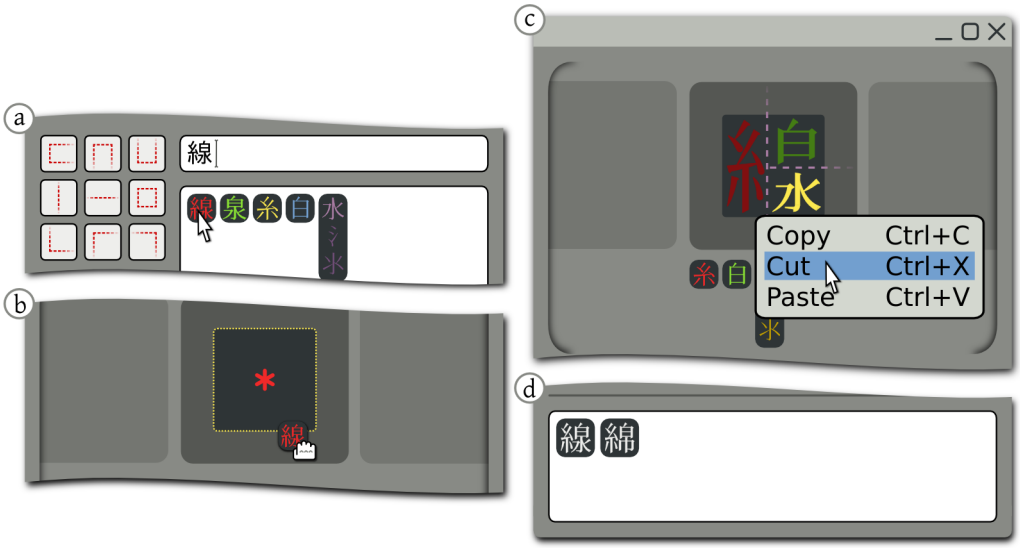


Illustration 3.7: Removing a part of the kanji 線to search for 綿

A final example presented here may eventually prove to be the most common way of using the *Kanji Navigator*. In the text sample from *I am a Cat* (p. 4) we encountered the rare kanji 譫 Assuming an intermediate learner of Japanese, the user might recognise that although he does not know its pronunciation, he does know that it appears to consist of 言and what looks like 虚 Simply entering both kanji into the argument input field and dragging the listed arguments not on, but directly below the kanji query character area should be sufficient in most cases; the number of real Chinese characters that consist of at least both these

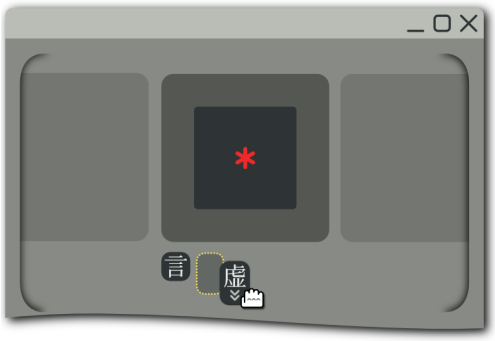


Illustration 3.8: Searching for the kanji 譫seen in Natsume Sōseki's *I am a Cat*

components in some spatial configuration is limited[35]. Illustration 3.8 shows the GUI just as the user drags in the second argument. The area beneath the kanji query character area lights up to indicate that it is a valid location to drop the argument. Notice that the character 虚in the user input in this example does not have the exact same shape as the right component of 譟 In these cases minute orthographic differences such as the difference between 虚and 虚– more specifically, the bottom-right part of those components, 业 and 业 – should initially be treated as variants of the same component. This is somewhat different than the *water* and *foot* variants seen above, as the difference is much more subtle, and tends to occur exclusively outside of the Jōyō Kanji chart. In this case too the user can decide to select one specific variant and use that in his query, but by default such variant shapes are hidden from sight — the 虚 argument in illustration 3.8 can be expanded to show the 虚variant by clicking the double chevron (next to the "hand" dragging the argument, directly below 虚.

3.3 Earlier designs sharing similarities

The notion of teaching learners about the components and structure of kanji in a more conscious way is not completely new. Some limited experimental proof-of-concept implementations exist. Although these implementations are invariably limited to a smaller set of kanji and targeted at L2 learners at the beginner level, the encouraging results of the experiments these applications were used in warrant a brief mention of this line of research. In the early nineties Bhatia (1991) introduced Kanji HyperCard; an experimental application that highlighted the top-level split of kanji from the Kyōiku Kanji set similar to the *position* column in table 2.2. A similar approach was taken by Nozaki & Ichikawa (1997). Unfortunately, while both initiatives proved that to some extent, the extra focus on teaching sub-character spatial arrangements and knowledge of the character's components contributed to the kanji retention and recollection abilities of

[35] Again, a cursory search of the data from the Kanji Database project shows that there is in fact only one matching character.

the subjects, the actual software created for neither of the above appears to be available on-line today.

Chapter 4

Software implementation

The designs presented in this thesis require innovative solutions for a number of practical problems; the kanji navigator in particular cannot function without a back-end database to tell it not only how to draw kanji, but also how to split it up into its separate, logical components — if any. Needless to say, the scope of such a resource necessitates relying on other resources to provide this data. Fortunately such resources exist; they are introduced in this chapter. Although the software proposed is still in the planning phase, a brief explanation of the chosen approach should help to shed some light on the technical issues that exist.

4.1 How to draw a Chinese character, *any* Chinese character

The idea of having any kanji — or indeed, any Chinese character from any of the CJKV cultures — available in a digital environment is appealing, but this requires the availability of computer fonts capable of displaying them. For individual, well-demarcated sets of characters such as the 6355 kanji in JIS standard `X 0208`, fonts are readily available — commercially, as well as for free under permissive licences — but rarer kanji are often missing; the kanji 譃(a rare character meaning *deception*, used in the Natsume Sōseki sample in illustration 1.3) for example,

is not included in the popular JIS X 0208[36], and is thus often omitted from fonts. Still, even if a good-looking, free[37] font was available, the information on the glyphs it contains is limited to their vector outlines; there is no indication of where a component begins, or ends. However, the design proposed in chapter 3 depends on having a widget available that knows quite a lot more about the characters it needs to display and allow interaction with. Not only does it need to be able to reproduce all these Chinese characters, it also has to be aware of the composition of a character's constituent elements, in order to fulfill the requirements for this widget. Practically speaking, it needs a resource that tells it how to draw Chinese characters, preferably one component at a time, and a resource that tells it what its constituent components are.

4.1.1 GlyphWiki

GlyphWiki[38] is a collaborative project that aims to provide the data necessary to create a font supporting the whole range of Chinese characters as defined in the Unicode standard, as well as other character sets (Kamichi 2006). GlyphWiki was born from the desire to allow anyone to implement existing Chinese characters, or even create their own kanji, without extensive experience in software engineering or typeface design being a prerequisite, and to provide an environment to store and retrieve these characters (Kamichi 2003, p. 85). In a sense, GlyphWiki employs a philosophy similar to that of the popular free encyclopaedia *Wikipedia*; the collaborative nature of this project allows it to grow at an impressive rate[39], and the principle of "many eyes" helps identify and correct faulty data. As of

[36] It does occur in the much more comprehensive JIS X 0213.

[37] Free as in *libre* as well as *gratis*, implying availability under a permissive free software licence.

[38] GlyphWiki can be accessed on-line at http://www.glyphwiki.org.

[39] In April of 2008 the Unicode Consortium proposed the addition of 4149 additional Chinese characters in the form of "CJK Unified Ideographs Extension C". In January 2009, a volunteer study meeting was held in Tokyo, explaining the basic design principles and methods for registering new characters in GlyphWiki. A mere two months later the entirety of this new extension was already implemented. The fruits of this labour — the Hanazono Minchō font automatically generated from GlyphWiki data — can be downloaded in the form of a TrueType font at http://fonts.jp/hanazono. Source: personal participation of the author.

July 2009, GlyphWiki data contains glyph definitions for well over 50000 Chinese characters, including all of the kanji part of JIS X 0208, X 0212 and X 0213.

As explained in Kamichi (2003), character shape definitions are stored at GlyphWiki on a per-line basis, consisting of colon-separated strings of up to eleven whole number values. Sequentially, these eleven values represent line-type (such as, *straight*, *curved*, *hooked*, etcetera), start and end decoration, start coordinate (coordinates are represented using two values), optionally up to two auxiliary coordinates, and the end coordinate. The coordinates are placed on an imaginary canvas with its origin (0, 0) in the top-left corner, and the bottom-right corner located at (200, 200). For the kanji 木(*tree*) for example, the data is as follows:

ⓐ 1 : 0 : 0 : 21 : 59 : 184 : 59
ⓑ 1 : 0 : 0 : 100 : 16 : 100 : 187
ⓒ 2 : 32 : 7 : 96 : 59 : 72 : 129 : 11 : 174
ⓓ 2 : 7 : 0 : 105 : 59 : 132 : 126 : 178 : 155

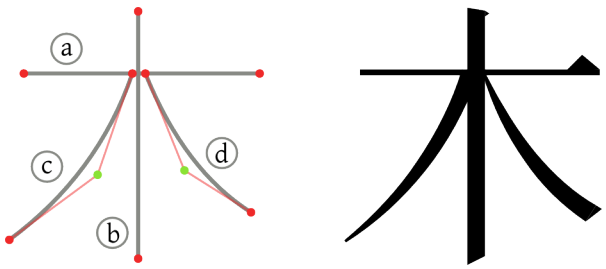


Illustration 4.1: Chinese character 木(*tree*) rendered using GlyphWiki data (stroke skeleton (left) and decorated minchō style output)

A more in-depth explanation of this notation falls beyond the scope of this thesis, but it suffices to note that the GlyphWiki data does not define the exact outline shapes of glyphs, rather, it defines its stroke skeleton. In illustration 4.1 this data for 木is visualised — the circles represent the coordinates defined therein[40] — with the four lines of data corresponding to the lines marked with

[40] The red circles are start and end coordinates. For the two curved lines, the green circle connected to the start and end coordinates by thin helper lines defines that lines curvature (mathematically speaking, this is the control point of a quadratic Bézier curve).

the same letter[41]. To actually draw the character, a drawing engine is required. For GlyphWiki, this functionality is provided by *KAGE/Engine* (Kamichi 2003), which uses decorating algorithms to draw the glyph using the common minchō[42] style and outputs vector outlines, but optionally other drawing engines could be used as well. The end result can be downloaded as raster images or vector outlines, or — provided the glyph is part of the Unicode specification — as a font-file together with other glyphs. The decorated kanji 木on the right of illustration 4.1 is the output of *KAGE/Engine*.

Use of GlyphWiki output throughout this document

Unless indicated otherwise, most of the Chinese character shapes used in the illustrations in this document use the vector outline output of glyphs defined at GlyphWiki[43]. With the high quality of these glyphs, and the permissive licensing specifically permitting alteration, GlyphWiki is a valuable resource for which the present author wishes to express his gratitude.

[41] It is also possible to reference other characters registered at GlyphWiki; for instance, the glyph outline for the complex character 譶s defined as:

```
99：0：0： 3 ：6：137：201：baseparts-gonben
99：0：0：57：7：194：198：baseparts-munashiib
```

Here, `baseparts-gonben` and `baseparts-munashiib` act as references to the glyph definitions for 言(specifically, in the form used when placed on the left side of a character) and 虛 The fourth, fifth, sixth, and seventh value together form the coordinates of the component in the whole, so for example the 言(`baseparts-gonben`) component is in this case located within an imaginary rectangle starting at (3, 6) and ending at (137, 201).

[42] Similar to the serif typefaces for Roman letters, minchō typefaces are characterised by the use of design features called serifs; the triangular shape at the right end of the horizontal stroke in 木(the 一) is such an element, called *uroko* (fish scale) in Japanese.

[43] Because the vector output downloadable from the GlyphWiki website splits up the kanji mostly at the stroke-level, colourizing or otherwise visually highlighting components is quite simple. This is not nearly as easy if the glyph outline extracted from a free software font is used, because then the components that overlap or touch are fused into one outline, frustrating visual separation.

4.1.2 CHISE/Kanji Database and the Unihan Database

With GlyphWiki, the bare minimum of data necessary to algorithmically draw Chinese characters is available, but it carries only limited information on the components and structure of complex kanji. Kanji such as 譳above represent fairly simple cases, with a clear-cut left-right split easily implemented by referencing existing components, but more complex cases exist, where the components are defined by specifying each line separately rather than using references to whole components. Furthermore, the data provides no indication of the abstract spatial layout — information needed to execute the sub-character queries proposed in chapter 3.

IDS data from CHISE/Kanji Database

For this type of information the ideographic description sequences (IDSs) from the Kanji Database Project (Kawabata, accessed 16th July 2009) may proof useful. The Kanji Database Project is a branch of the CHaracter Information Service Environment (CHISE) (Morioka 2005) dataset, with an added distinction made between the orthographical variations of the unified Chinese characters in the Unicode standard. A well-known example of such glyph shape variations are the variant characters for *bone* encoded at codepoint U+9AA8, which are considered *unifiable* according to the Han unification rules set out in the Unicode standard (The Unicode Consortium 2007, p. 417–421), but — when comparing the different styles of writing them in the various CJKV cultures — visually distinct (illustration 4.2).

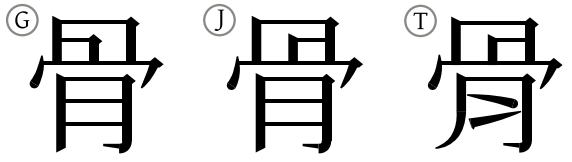


Illustration 4.2: Chinese character for *bone* (U+9AA8), as used in China (G), Japan (J), and Taiwan (T)

Where possible, the IDS data from the Kanji Database Project uses other characters in the Unicode standard to define the spatial configuration of complex

kanji — in cases where no Unicode character exists for a specific component, additional glyphs from an external glyph set are used instead[44]. The spatial relation between two or three components is expressed using ideographic description characters[45] (The Unicode Consortium 2007, p. 427–430) — these indicate how a character or component is split up; ⿱ for instance indicates two components stacked vertically. Taking 峠for an example again, the relevant IDS is as shown in table 4.1. Note that unlike 卡, the characters 山 上and 下cannot be decomposed any further at the component-level[46].

Unicode codepoint	Character	IDS
U+5CE0	峠	⿰ 山卡
U+5C71	山	*none*
U-000209D7	卡	⿱ 上下
U+4E0A	上	*none*
U+4E0B	下	*none*

Table 4.1: The IDS data for the kanji 峠

A combined glyph information database

The availability of these two resources does not guarantee that all the information necessary to distinguish Chinese character components is present. In most cases, GlyphWiki lines correspond to the traditional character strokes, but in some cases more than one GlyphWiki line is used to describe a single stroke[47]. This means that an effort to separate GlyphWiki lines into component groups, cannot depend solely on the stroke counts for components to separate GlyphWiki definitions that do not use a reference to another kanji or component. Fortunately, glyphs registered at GlyphWiki generally do define the components in

[44] This additional glyph set consists of a set of characters defined by the Chinese Document Processing Lab in Taiwan. These glyphs can be mapped to the private use area in Unicode by means of a mathematical formula (Kawabata, accessed 16th July 2009), and can thus be used in a Unicode text document, provided a font containing these glyphs is installed.

[45] Unicode character range U+2FF0 – U+2FFB.

[46] Although they can of course be dissected into strokes.

[47] For example, the 口(*mouth*) component is defined by four GlyphWiki lines, whereas its traditional stroke count is three.

their correct order. 峠for example is defined as a reference to one of the variant forms for 山 and six lines — the first three signifying 上, and the rest 下. One approach under consideration to link the two resources, is a tunable heuristic algorithm[48] capable of weighing the various ways of splitting up a GlyphWiki glyph definition into its components, and reporting any ambiguous cases for further review and/or fine-tuning of the algorithm. To implement such an algorithm, basic Chinese character knowledge such as the stroke count is also necessary; this information is provided by the Unihan Database (The Unicode Consortium 2007, p. 131).

4.2 GlyphWiki Drawfont Tool

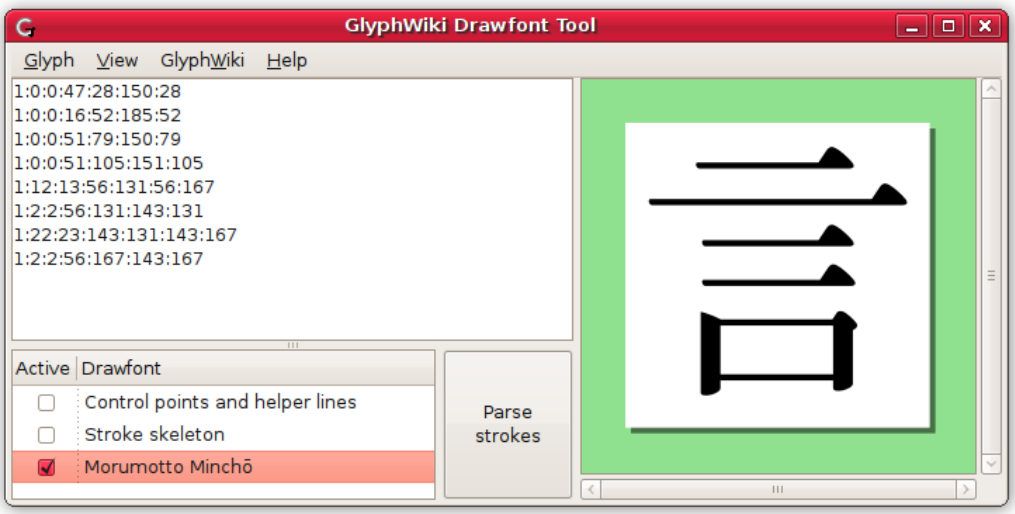


Illustration 4.3: *GlyphWiki Drawfont Tool* showing basic drawing engine functionality for the GlyphWiki definition of 言(*word*, *speech*)

Experimental implementation of the basic functionality needed for the Kanji Navigator has started in the form of *GlyphWiki Drawfont Tool*[49]. The short term

[48] An algorithm that gives the correct results for the vast majority of the data input, but may not find the optimum solution for all cases, necessitating some user intervention.

[49] Source code and public releases are available at `http://gwdrawfonttool.sf.net`. Current releases are likely to be of interest only to other software developers. The software is released

goal is the development of a reusable GTK+ widget capable of drawing kanji defined at the GlyphWiki project, and highlighting their components. Similar to *KAGE/Engine*, the focus lies on rendering the glyphs using a Minchō typeface, although it is feasible to provide a Gothic typeface as well at some future point. In addition to diagnostic "drawfonts" (decorating algorithms) showing the bare skeleton defined by the GlyphWiki data — an example of which is the stroke skeleton shown in illustration 4.1 — an experimental minchō typeface called *Morumotto Minchō* is being developed within the context of *GlyphWiki Drawfont Tool*. Illustration 4.3 shows this application rendering a basic Chinese character using the *Morumotto Minchō* drawfont.

4.3 Software environment

Because ultimately the goal is to attain cross-platform availability, the programming language and supporting software chosen[50] are available for all three major desktop OSs — that is, Apple's Mac OS X, the various GNU/Linux distributions, and Micosoft's Windows. All of the resources chosen (including GlyphWiki and CHISE/Kanji Database) share one important characteristic; they can all be classified as free and open-source software (FOSS) — software licenced under permissive terms, explicitly allowing redistribution, adaptation, and access to the source code[51]. Similarly, *GlyphWiki Drawfont Tool* and the future implementation of the ideas presented in this thesis too, respectively are, or will be, licenced

under the terms of the GNU General Public Licence (GPL) and is currently targeted at the GNU/Linux OS (eventual expansion to other OSs is planned).

[50] Currently the software environment used for GlyphWiki Drawfont Tool consists of Python, GTK+, Gettext, Cairo and SQLite. Practical reasons for this selection include the legibility of Python (which can be considered advantageous for source code that is meant to be read by others as well) and the versatility of the GTK+ widget toolkit (especially the ease of using GUI translations for other languages in combination with Gettext; the GUI of *GlyphWiki Drawfont Tool*, for example, can be used in Dutch, English, or Japanese, depending on the user's preferences).

[51] This is in the author's opinion very relevant for academic research — even if the software part of an experiment was only used to support a hypothesis and would be considered obsolete or impractical by any standard, details about its implementation can provide valuable clues for future research. The *Kanji Finder* software used by Bhatia (1991), for example, can no longer be retrieved on-line — let alone its source code — even though these may be considered crucial to the research described in the accompanying article. Without the software, the research is no longer *reproducible* — certainly one of the tenets of any research relying on digital resources.

under a free software licence, such as the GPL[52] for the source code, and more permissive licences for any other resources.

[52] More information on the GPL can be found on-line at `http://www.gnu.org/copyleft/gpl.html`.

Chapter 5

Concluding remarks

The psycholinguistic theoretical background presented in chapter 2 and the conceptual designs proposed in chapter 3 warrant further research into the applicability of methods employing direct access to the sub-character features of Chinese characters in a digital environment — both through visualisation (highlighting the various components and their functions), and manipulation (user-created sub-character queries). Furthermore, if the notion of sub-character queries proves successful within the context of an electronic dictionary for the Japanese language, it may be worthwhile to explore the possibility of expanding this concept to the IME-level, allowing users to enter rare, unfamiliar, or otherwise hard to enter kanji using a derivative of the *Kanji Navigator*. Following this approach, in due course the usefulness of the ideas presented in this thesis for L1 users should be explored as well.

5.1 Applicability to other CJKV languages

Although this research focuses on Chinese characters as used in Japanese, the designs presented in this thesis may be — to some extent — applicable to other CJKV languages using these logographic characters as well. Unfortunately, just as this thesis builds upon research written in Japanese and English, any in-depth research on Chinese character usage within the context of those cultures, would require delving into research written in the languages used in those cultures —

Chinese, or Korean, for example — as well as practical experience in learning those languages. Global initiatives, such as the encoding of Chinese characters via the Unicode Consortium, help bridge the digital divide between the various CJKV cultures, but language and cultural barriers are unlikely to disappear in the near future. Yet, free and open-source — and thus globally available — approaches to Chinese character processing may ultimately lead to new, commonly available ways of using Chinese characters in a digital environment.

5.2 Lacunae in the present research

As shown in chapter 2, research focussing on the cognitive processing of kanji by L2 learners from a non-kanji cultural background exists, but it is severely limited in number. The body of research that does exist often bases itself on results gathered from studying learners in the beginning or lower-intermediate phases of study. For a broader perspective, studies that include more advanced students of Japanese as well, could help to more accurately identify the similarities and differences with native learners, as well as pinpoint the weaknesses L2 learners experience in acquiring and using their kanji vocabulary. Likewise, a comprehensive study into the usage and desired functionality of (electronic) dictionaries by L2 learners of Japanese would greatly benefit the development of a future dictionary user interface, but falls beyond the scope of the present research. What do users miss in the current generation of Japanese dictionaries and how do they rate the usefulness of the ideas presented in this thesis? Guided by HCI methodology, eventually user questionnaires and interviews aimed at users of existing dictionary software and early experimental versions of the design proposed in this thesis, could help identify annoyances, lacking functionality, and common usage patterns, allowing further refinement of the designs in chapter 3.

The present research lacks a further in-depth exploration of the various language acquisition theories that exist. For the purposes of this thesis — that is, providing the rationale for a GUI design allowing interaction with kanji subcharacter components — only a very generalised overview of some of the differences between L1 and L2 learners and the issues L2 learners face during the pro-

cess of learning Japanese is presented. There are a number of modern theories concerning the difference between native and non-native users of a language, most of which go well beyond the traditional *learning versus acquisition* view, but a comprehensive treatise of this topic would fall well beyond the scope of this thesis.

Because this thesis presents only the theoretical background together with the initial design concepts, coverage of the technical implementation details is severely limited. There are quite a number of issues that have to be overcome before these designs can become reality. For example, in illustration 3.5 the 足 (*foot*) component inserted by the user automatically turns into a variant shape when inserted into the query; namely 𧾷, the form it generally takes when used on the left-side of a character. Such transformations are desirable as they mimic actual usage, and it can be automated, but this requires a fairly extensive database of component variations and usage rules. With the combination of CHISE and the Kanji Database, the Unihan Database, and GlyphWiki, this information is freely available, and such a database might be distilled from it, but there is as of yet a lot of ground to be covered.

5.3 The road ahead: refining the design

The conceptual design phase is but the first step in the usability life cycle of the final product (Mayhew 2008). From a technical perspective, the next step is to build the supporting architecture — the main component of which is the database that holds the information on how to draw kanji, how to split it up in its constituent components and what components fulfill semantic and phonetic functions, as well as all the data a user would expect in a kanji dictionary, such as readings, dictionary references, meanings, etcetera — and to implement a rough first implementation, proof-of-concept of the *Kanji Navigator* component, building upon the work done with *GlyphWiki Drawfont Tool*.

From thereon, practical testing with interested users and developers is needed in order to iteratively come to a more polished and detailed design, and subsequently a proper implementation. Should these GUI concepts proof successful, the long term goal will be to make the software available on as wide a

range of platforms as possible. If circumstances permit it, a working implementation could be used as part of psycholinguistic experiments as well, helping to shed some further light on the way L2 learners (subconsciously) deal with kanji, possibly addressing some of the lacunae identified above. In the short term however, user involvement is the key to moving beyond the experimental phase, in line with the open-source "release early, release often" mantra, the creation of a usable prototype — regardless of its faults and limitations — is the next objective.

Bibliography

Apel, U. & Quint, J. (2004), Building a graphetic dictionary for japanese kanji—character look up based on brush strokes or stroke groups, *in* 'Proceedings of the 20th International Conference on Computational Linguistics'.

Benson, C., Clark, B. & Nickell, S. (2008), *GNOME Human Interface Guidelines 2.2*, The GNOME Project.
URL: http://library.gnome.org/devel/hig-book/stable/

Bhatia, A. (1991), 'Kanji retrieval by recursive location of elements using hypercard', *CALICO Journal* **9**(2), 4–25.
URL: https://www.calico.org/html/article_525.pdf

Chikamatsu, N. (2005), L2 japanese kanji memory and retrieval: An experiment on the tip-of-the-pen (TOP) phenomenon, *in* V. J. Cook & B. Bassetti, eds, 'Second Language Writing Systems', Multilingual Matters, pp. 71–96.

Gottlieb, N. (2000), *Word-Processing Technology in Japan – Kanji and the Keyboard*, Curzon Press, Richmond.

Hadamitzky, W. & Spahn, M. (1997), *Kanji & Kana*, first revised edn, Tuttle Publishing, Tokyo.

Heisig, J. W. (2007), *Remembering the Kanji 1*, fifth edn, University of Hawai'i Press, Honolulu.

Hirose, H. (1999), 'Kanji no ninchi ni okeru keitai yōso no kinō ni tsuite no ikkōsatsu [A discussion on functions of graphemic components on recog-

nition of kanji]', *Ryūkyū Daigaku Kyōiku Gakubu Kiyō* **55**, 259–274.
URL: http://ir.lib.u-ryukyu.ac.jp/handle/123456789/2137

Hoek, J. D. (2007), A look at the future of the denshi jisho – designing the next generation of digital japanese dictionary software, Bachelors's thesis, Leiden University.
URL: http://handle.jeroenhoek.nl/hoek_2007

Ishikawa, S. (2004), 'Kyōikuteki shiten ni motozuku denshi jisho no yūzā intāfēsu saikō - kensaku gamen dezain no kentō [A reconsideration of the user interface of digital dictionaries from an educational standpoint]', *Gengo Bunka Gakkai Ronshū* **22**, 53–68.

Itō, K. (1979), 'Keisei moji to kanji shidō [Semasio-phonetic characters and kanji guidance]', *Gengo Seikatsu* **326**, 68–81.

Kamichi, K. (2003), Kage – an automatic glyph generating engine for large character code set, *in* 'Proceedings of the Glyph and Typesetting Workshop', Kyoto University Institute for Research in Humanities, pp. 85–92.

Kamichi, K. (2006), 'GlyphWiki – kaihōgata fonto kaihatsu kankyō no kōchiku ni mukete [GlyphWiki – towards the construction of an open font development environment]', *Kanji Bunken Jōhō Shori Kenkyū* **7**, 12–18.

Kawabata, T. (accessed 16th July 2009), 'Kanji Dētabēsu Purojekuto [Kanji Database Project]', project website.
URL: http://kanji-database.sourceforge.net/

Kess, J. F. & Miyamoto, T. (1999), *The Japanese Mental Lexicon: Psycholinguistic Studies of Kana and Kanji processing*, John Benjamins Publishing Company, Philadelphia/Amsterdam.

Mayhew, D. J. (2008), Requirements specifications within the usability engineering lifecycle, *in* A. Sears & J. A. Jacko, eds, 'The Human-Computer Interaction Handbook', Lawrence Erlbaum Associates, New York, pp. 917–926.

Mori, Y. (2003), 'The roles of context and word morphology in learning new kanji words', *The Modern Language Journal* **87**(3), 404–420.

Morioka, T. (2005), Character processing based on character ontology, *in* 'Nicchū Kyōdō Shinpojiumu - Kanji Bunken Shiryōkoteki Shingijutsu'.

Nelson, A. N. & Haig, J. H. (1997), *The New Nelson Japanese–English Character Dictionary*, Tuttle Publishing.

Nomura, H. (2007), 'Konjaku Mojikyō, Chō-Kanji Kensaku [Konjaku Mojikyō, Chō-Kanji Search]', *Japan Association for East Asian Text Processing* **8**, 140–143.

Nozaki, H. & Ichikawa, S. (1997), 'Kanji gakushū shien shisutemu – kanji no kōzō rikai to suji undō kankaku no kakutoku [Kanji learning support system – the acquisition of kanji structure comprehension and muscle movement sensitivity]', *Nihon Kyōiku Kōgakkai Ronbunshi* **21**(1), 25–35.

Saito, H., Kawakami, M. & Masuda, H. (1995*a*), 'Kanji kōsei ni okeru buhin (bushu) no shutsugen hindo hyō [Occurence frequency chart of the parts (radicals) found in kanji structures]', *Jōhō Bunka Kenkyū* **1**, 113–134.

Saito, H., Kawakami, M. & Masuda, H. (1995*b*), 'Kanji kōsei ni okeru buhin (bushu) – on'in taiō hyō [Parts (radicals) found in kanji structures – phoneme correspondence chart]', *Jōhō Bunka Kenkyū* **2**, 89–115.

Saito, H., Masuda, H. & Kawakami, M. (1998), 'Form and sound similarity effects in kanji recognition', *Reading and Writing* **10**(3), 323–357.

Saito, H., Yamazaki, O. & Masuda, H. (2002), 'The effect of number of kanji radical companions in character activation with a multi-radical-display task', *Brain and Language* **81**(1-3), 501–508.

Suzuki, Y. (2007), 'A note on kanji-education', *Bulletin of Center for Japanese Language* **20**, 53–70.
URL: http://hdl.handle.net/2065/26475

Takebe, Y. (1989), *Kanji no Oshiekata [A Method of Teaching Kanji]*, ALC Press Incorperated, Tokyo.

Tamaoka, K. (2005), 'The effect of morphemic homophony on the processing of japanese two-kanji compound words', *Reading and Writing* **18**(4), 281–302.

Tamaoka, K. & Makioka, S. (2004), 'New figures for a web-accessible database of the 1,945 basic japanese kanji, fourth edition', *Behavior Research Methods, Instruments, & Computers* **36**(3), 548–558.

Tamaoka, K. & Yamada, H. (2000), 'The effects of stroke order and radicals on the knowledge of japanese kanji orthography, phonology and semantics', *Psychologia: an international journal of psychology in the Orient* **43**(3), 199–209.

The Unicode Consortium (2007), *The Unicode Standard, Version 5.1.0*, Addison-Wesley, Boston.
URL: http://www.unicode.org/versions/Unicode5.1.0/

Tokuhiro, Y. (2003), 'Kanji ninchi shori kara mita kōkateki kanji shūtokuhō no kenkyū – sōgo ketsugōgata gainen chizu sakusei no kokoromi [A study of efficient kanji acquisition methods from a cognitive viewpoint – experimenting with the creation of an interlinked concept map]', *Waseda University Nihongo Kyōiku Kenkyū* **2**, 151–176.
URL: http://hdl.handle.net/2065/3502

Watzman, S. & Re, M. (2008), Visual design: Principles for usable interfaces, *in* A. Sears & J. A. Jacko, eds, 'The Human-Computer Interaction Handbook', Lawrence Erlbaum Associates, New York, pp. 329–354.

Wei, Q., Ihara, A., Hayakawa, T., Murata, T., Matumoto, E. & Fujimaki, N. (2007), 'Phonological influences on lexicosemantic processing of kanji words', *NeuroReport* **18**(17), 1775–1780.

Glossary

CHISE *CHaracter Information Service Environment*.

CJKV *China, Japan, Korea and Vietnam*. The cultures where Chinese characters are, or were, used. China implies both mainland China and Taiwan, Korea both North and South Korea. Because Vietnam no longer uses Chinese characters in its modern language, the term *CJK* is sometimes used instead.

CSS *Cascading StyleSheets*. Reusable instructions for describing the style (such as colours, font faces, font size, etc.) of elements in a markup language (such as HTML), used to separate layout and formatting from actual content.

display resolution The number of columns and rows in a monitor's pixel grid.

FOSS *free and open-source software*.

glyph A specific interpretation of the shape of a character, such as the vector outline of a character as defined by a particular computer font.

GPL *GNU General Public Licence*.

GUI *graphical user interface*. Essentially, the part of a software application visible to the user, often comprised of a number widgets placed inside one or more windows.

HCI *Human-Computer Interaction*. An area of research focussing on the human side of the interaction between man and computer.

IDS *ideographic description sequence*.

IME *input method editor*. In a software environment this tool is used for inputting text in scripts not directly supported by the input device (such as Chinese, Japanese or Korean). For instance, by converting a word input as a transliteration using the Roman letters found on a keyboard to the desired script. If multiple possibilities exist it can offer a list of options to the user; e.g. for the Japanese noun "Nioi" (smell/fragrance), 匂い, 臭い, におい, and ニオイ might be suggested. Sometimes simply referred to as *input method*.

JIS *Japanese Industrial Standards*. JIS standards with a name starting with `JIS X`are information processing standards, such as the definitions of the various character sets (e.g. `JIS X 0208`).

L2 *Second language*. A language typically acquired after puberty, thus without the benefits learning a language at a young age brings. One's native language, or *mother tongue*, and any fluently spoken language acquired before adulthood are sometimes called *first languages (L1)*.

mora Linguistic term for a unit of sound with specific properties (timing, pitch, etc.). A feature of the Japanese language is the equal timing of morae; all morae are roughly pronounced for the same duration, with the bi-moraic long vowel sounds being held for twice the duration of a mono-moraic short vowel.

netbook New generation of portable computers bigger than PDAs and smartphones, yet significantly smaller than laptops. The form factor is usually similar to that of a laptop.

OS *operating system*. The core software of a digital device upon which other software runs. Well-known examples are variants of GNU/Linux, Mac OS X and Windows for the personal computer, but smartphones, PDAs and PEDs too have one, often a derivate of the aforementioned OSs. Whether or not an application can run on any one OS depends on the software itself (does a version compatible with this OS exist) and the availability of software libraries used by the application.

PDA *personal digital assistant*. Class of portable devices initially primarily designed as digital organisers (calendar, address book, notebook, etc.), but more recently most of the models sold double as smartphones. Most modern PDAs use an OS that allows the user to freely install software, making them suitable as PED replacement, provided dictionary software exists for its platform.

PED *portable electronic dictionary*. Dedicated portable devices loaded with digital dictionaries, such as Canon's Wordtank series. In shape comparable to the smaller netbooks, but not nearly as powerful. They are functionally limited to the dictionary software pre-installed by the manufacturer, although extra dictionaries can often be added afterwards provided they come in a digital format suitable for the device.

raster image *See:* vector graphics.

smartphone Mobile phones with PDA-like functionality, essentially turning the device into a versatile portable computer.

stylus A small pen or pencil-shaped utensil with a rounded tip, used as an alternative to a computer mouse for devices with touch-screens (PEDs, PDAs or netbooks). With a stylus, the user can effectively "write" on the screen of such a device and have the computer recognise the characters.

syllabary Set of written characters where each character represents a single syllable.

vector graphics In a computing environment images can basically be defined in two ways: as raster images (describing the colour of each point (*pixel*) on a rectangular grid (*raster*)) or as vector-based images (definitions of geometrical primitives (such as polygons and curved lines) specifying only the coordinates needed for the computer to mathematically draw them at any size). The latter is used for images that need to be scalable without loss of information (i.e., the image looking blurred).

widget In a software environment, widgets are components that enable user interaction or provide information to the user, such as buttons, scrollbars and text fields, but also more complex components such as drawing canvas in a graphics editing application.