AN ANALYSIS OF THE EFFICIENCY OF EXISTING KANJI INDEXES AND DEVELOPMENT OF A CODING-BASED INDEX

Galina N. VOROBEVA[*]
Kyrgyz National University
g.vorobyova@yahoo.com

Victor M. VOROBEV
Kyrgyz National University
v.vorobyov@yahoo.com

Abstract

Considering the problems faced by learners of Japanese from non-kanji background, the present paper discusses the characteristics of 15 existing kanji dictionary indexes. In order to compare the relative efficiency of these indexes, the concept of selectivity is defined, and the selectivity coefficient of the kanji indexes is computed and compared. Furthermore, new indexes developed by the present authors and based on an alphabet code, a symbol code, a semantic code, and a radical-and-stroke-number code are presented and their use and efficiency are explained.

Keywords

kanji index; search; efficiency index; selectivity; kanji coding

Izvleček

Pričujoči članek obravnava lastnosti 15 obstoječih kazal v slovarjih kitajskih pismenk. Da bi primerjali relativno učinkovitost teh kazal, definiramo koncept izbirnosti ter izračunamo in primerjamo koeficient izbirnosti teh kazal. Nadalje predlagamo in predstavljamo rabo in učinkovitost novih kazal, ki smo jih osnovali na kodiranju z latiničnimi črkami, simbolnem kodiranju, semantičnem kodiranju in kodiranju na osnovi pomenskega ključa in števila potez.

Ključne besede

kazalo kitajskih pismenk; iskanje; indeks učinkovitosti; selektivnost; kodiranje kitajskih pismenk

[*] Translated by Boštjan Bertalanič, Andrej Bekeš and Kristina Hmeljak Sangawa

1. Background of the study

The most commonly known indexes used to look up kanji (Chinese characters) in character dictionaries are the radical index (部首索引 *bushu-sakuin*), the stroke-number index (総画数索引 *sōkakusuu-sakuin*) and the readings index (音訓索引 *onkun-sakuin*). The radical index is used when searching kanji by means of their radical; it consists of a list of all radicals arranged by increasing number of strokes, where each radical is followed by a list of characters belonging to this radical, and each list of characters with the same radical is usually arranged by increasing number of strokes. The stroke number index is used to look up characters when their number of strokes is known; it consists of all characters contained in a dictionary, arranged by increasing total number of strokes. The readings index is used to look up characters by their reading, and consists of a list of all readings of the characters contained in the dictionary, usually arranged in standard *gojūon* kana order.

However, it is also generally known that learners from non-kanji background, i.e. learners who are not familiar with Chinese character writing, find it difficult to use traditional character dictionaries. When searching in a traditional radical index, it is sometimes difficult to determine which part of the character is to be considered its radical, as in the case of 巨 where 工 is the radical. Traditional ordering is complicated also because it does not classify characters consistently according to shape, but rather takes into account meaning, such as in the case of the characters 間 (“between”), 閉 (“close”), 開 (“open”) which are indexed under radical 門 (“gate”), while the character 問 (“question”) is indexed under radical 口 (“mouth”) and 聞 (“listen”) under radical 耳 (“ear”). The user should therefore already know in advance the meaning of a character in order to look it up, which is seldom the case. Indexes ordered by number of strokes are also troublesome to use, since we tend to make mistakes when counting, and there are many characters with the same number of strokes. In order to use reading indexes, on the other hand, the user needs to know the reading of a given character in order to look it up, but most users from a non-kanji background generally look up characters exactly because they do not know how to read them.

2. Previous research

Previous research has aimed at developing more efficient search methods. Both in the cultural sphere using Chinese characters and outside of it, diverse types of search methods have been developed and are being used besides the above mentioned and well known radical index, stroke index or readings index. To mention a few, other methods include the “five step arrangement kanji table” (五段排列漢字表 *godan hairetsu kanji hyō*) developed by the Russian researcher Rosenberg (1916), the Four corner method (四角号碼 *Sì jiǎo hào mǎ*) developed in China (Wang, 1925), the phonetic key index (音符 *onpu*) developed by Shiraishi (1971/1978), the index of katakana shapes, the initial stroke pattern index and the index of meaning symbols

developed by Kanō (1998), the index of stroke order patterns by Wakao and Hattori (1989), the index of character shapes by Sakano, Ikeda, Shinagawa, Tajima and Tokashiki (2009), the key words and primitive meanings index by Heisig (1977/2001), a radical index consistently based on shape by Hadamitzky and Spahn (1981), the system of kanji indexing by patterns (SKIP) developed by Halpern (1988), and the Kanji Fast Finder system by Matthews (2004). The authors of the present paper have also contributed to research on character indexing (Vorobeva, 2009, 2011).

3. Research aims

The goal of this paper is to 1) describe existing character search methods and analyse their efficiency; and 2) develop an efficient character search method based on a coding of character which appropriately expresses character form.

4. Research method

Given the variety of existing character indexes, we considered that an evaluation and comparative analysis of their efficiency was necessary. In order to compare the efficiency of the various character indexes, in this study we decided to use the concept of *selectivity* which expresses the processing efficiency of computer data. With reference to character indexes, we introduced the concept of *selectivity coefficient* and used it to compare and assess the efficiency of existing character indexes by calculating their selectivity coefficient.

We then built some new types of character indexes on the basis of character codes which accurately express character form. We constructed a database with four kinds of character codes for all new *Jōyō kanji*, sorted the data according to the order used in dictionary compiling, and developed four indexes: an alphabet code index, a symbol code index, a semantic code index and a radical and stroke code index.

Finally, we compared and evaluated the efficiency of these four code indexes.

5. Types and characteristics of existing character indexes

In the following paragraphs we introduce 12 existing types of character indexes, omitting the *on-kun* readings index, the radical index and the stroke-number index already described in the introduction.

5.1 Five step arrangement kanji table - 五段排列漢字表

The “graphic system” search method developed by the Russian researcher Vasil’ev (1867) and the “five step arrangement kanji table” developed by Rosenberg (1916) are generally known as the “Russian graphic system” (“Kod_Rozenberga”, 2012).

This is a method of arranging characters by form. Users only need to remember some simple rules and can use this method even when they cannot determine which part is the radical, have problems counting the number of strokes and do not know how to read the character. The method was developed by professor Vasilij Pavlovič Vasil´ev, a Chinese language scholar at Kazan University in Russia, who extracted 19 kinds of graphic elements or strokes (see Figure 1) which compose Chinese characters, and classified each character according to its last stroke, i.e. the stroke which is written last.

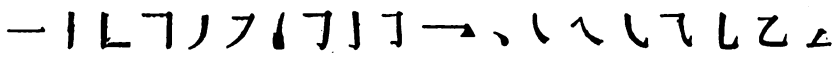


Figure 1: Vasile’v’s 19 graphic elements of Chinese characters

On the basis of these graphic elements, he compiled a new type of character dictionary with a unique character ordering and search method: the first Chinese-Russian dictionary, entitled Графическая система китайских иероглифов. Опыт первого китайско-русского словаря [*Grafičeskaâ sistema kitajskih ieroglifov. Opyt pervogo kitajsko-russkogo slovarâ. - “Graphic System of Chinese Characters. An attempt at the First Chinese-Russian Dictionary.”*] (Vasil’ev, 1867, see figure 2). This was the beginning of a far-reaching reform in dictionary structuring and organisation.

Figure 2: Vasil’ev’s *First Chinese-Russian Dictionary*

Two decades later, professor D. A. Pešurov at Saint Petersburg University published a Chinese-Russian dictionary entitled Китайско-русский словарь. по графической системе (*Kitajsko-russkij slovar´ po grafičeskoj sisteme* - *"Chinese-Russian Dictionary. Based on the Graphic System"*, Pešurov, 1891) on the basis of Vasil'ev's graphic method (see figure 3).


漢俄書法合壁字彙

КИТАЙСКО-РУССКІЙ

СЛОВАРЬ.

(ПО ГРАФИЧЕСКОЙ СИСТЕМѢ.)

Д. А. ПЕЩУРОВА.

ЦѢНА 2 РУБ.

САНКТПЕТЕРБУРГЪ.

1891.

Figure 3: Pešurov's Chinese-Russian Dictionary

The Russian graphic system was subsequently also used in a Japanese character dictionary, compiled by professor Otto Rosenberg of St. Petersburg University, who emphasised the importance of kanji learning stating as follows: "… especially for the person who is going to study the literary arts of Japan and Japanese civilisation of former ages, the knowledge of Chinese characters should become the core subject of learning"[1] (Rosenberg, 1916, p. 7). At the same time, he also discussed the difficulties to be faced when learning Chinese characters and using character dictionaries, and wrote the following comment regarding characters being listed according to their radicals: "I feel considerable inconvenience and difficulty, because Chinese characters do not possess an ordering such as alphabets[2]" (Rosenberg, 1916, p. 1).

On the basis of the graphic system developed by Vasil´ev, Rosenberg constructed a search system that was based on the shape of the characters and was completely different from the traditional indexes based on radicals, stroke number and readings, and used it to compile the 五段配列漢字典 (*Godan hairetsu kanjiten* - *"Five Step*

[1] 特に前代日本の文明，前代日本の文学芸術を知らんとする人に取りては，漢字の知識は，その主要科目たるべきなり。

[2] 非常なる不便と困難とを感じたり。それは主として漢字にアルファベットの如き順序なきによれるなり。

Arrangement Character Dictionary") a dictionary with a novel character arrangement and search system published in Japan (Rosenberg, 1916, see figure 4).

Figure 4: Rosenberg's 五段配列漢字典 (*Godan hairetsu kanjiten*, 1916)

The basic idea of the five-step arrangement search system is that "One look at the shape of the character [...] is enough. It can then immediately be found rapidly and reliably[3]" (Rosenberg, 1916, Explanatory notes 1).

Russian speakers are used to the notation in Cyrillic and Roman letters, and even in the case of Chinese characters they feel the need for a systematisation similar to the alphabet. Rosenberg took that into account when classifying and arranging characters according to the last written stroke. In contrast to Vasil'ev's system with 19 strokes (Vasil'ev, 1867), Rosenberg used 24 strokes and divided them into 5 groups. In order to implement a Chinese character ordering system based on stroke order, Rosenberg extracted 24 types of strokes, and grouped them into 5 categories, according to the direction in which they are written. Finally, he selected one stroke to represent each of the five groups, as shown in table 1 (Rosenberg, 1916, p. 20).

Table 1: Basic strokes in Rosenberg's system

Direction	Basic strokes
↗	丿
↘	乀
↙	丿
↓	丨

[3] 文字(中略)の形を, 一見したるのみにて, 十分なり。而して直ちに迅速確實に検出することを得べし。

→	一

Rosenberg exemplified the 5 types of strokes using the 5 strokes of the character 本, as shown in figure 5.

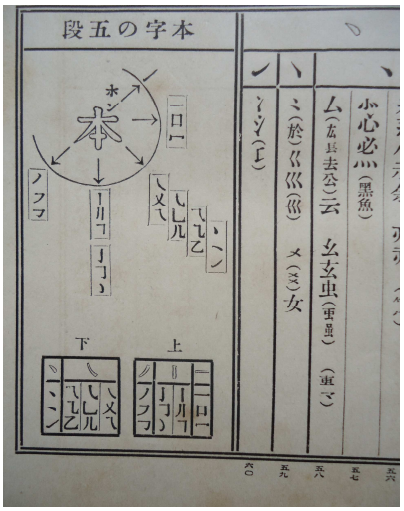


Figure 5: Rosenberg's five types of strokes, exemplified by the character 本

The Chinese character chart (漢字の字母表 *kanji no jibohyō*) in Rosenberg's (1916) dictionary is shown in figure 6, while figure 7 shows one page of the Five step arrangement Chinese character index (五段配列漢字表 *godan hairetsu kanjihyō*), and figure 8 shows one page of the dictionary's main entries.

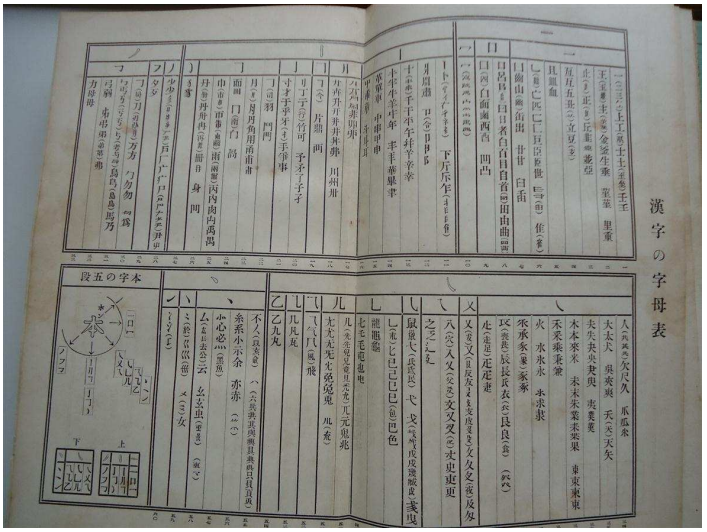


Figure 6: The Chinese character chart (漢字の字母表 *kanji no jibohyō*) in Rosenberg's dictionary

The Chinese character type chart (see figure 6) includes the five basic strokes, beneath them all 24 strokes classified by shape as belonging to one of the basic five strokes, and finally, beneath them, 567 character types (Chinese characters and character patterns) classified according to the 24 strokes and divided into 60 columns. Characters listed in the Five step arrangement Chinese character index (五段配列漢字表 *godan hairetsu kanjihyō*, see figure 7) are arranged according to the order of their strokes in the character chart. Figure 8 shows an example dictionary page.

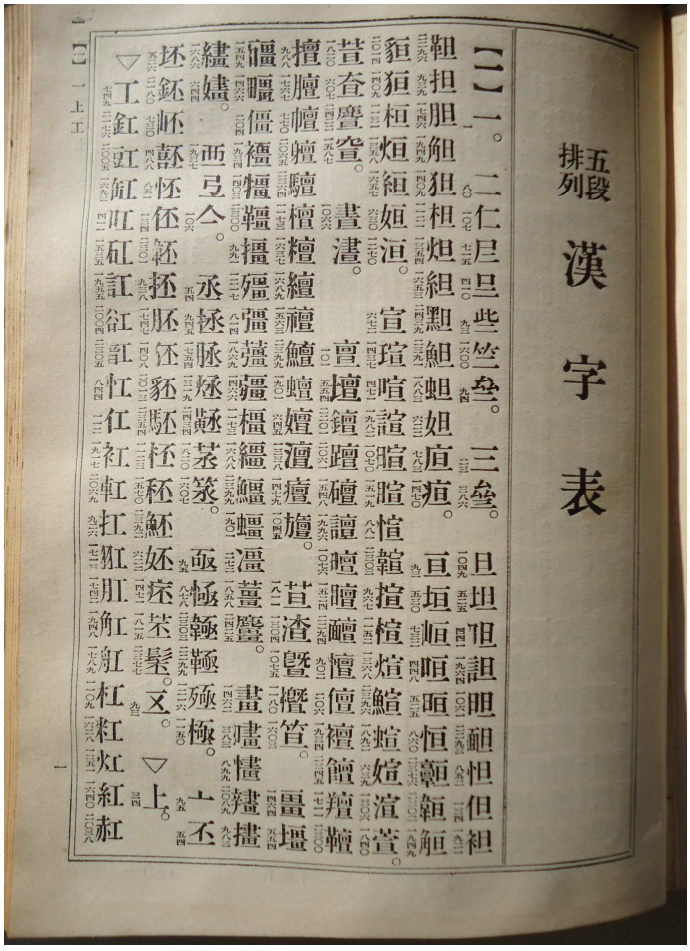


Figure 7: First page of the Five step arrangement Chinese character index (五段配列漢字表 *godan hairetsu kanjihyō*) in Rosenberg's dictionary

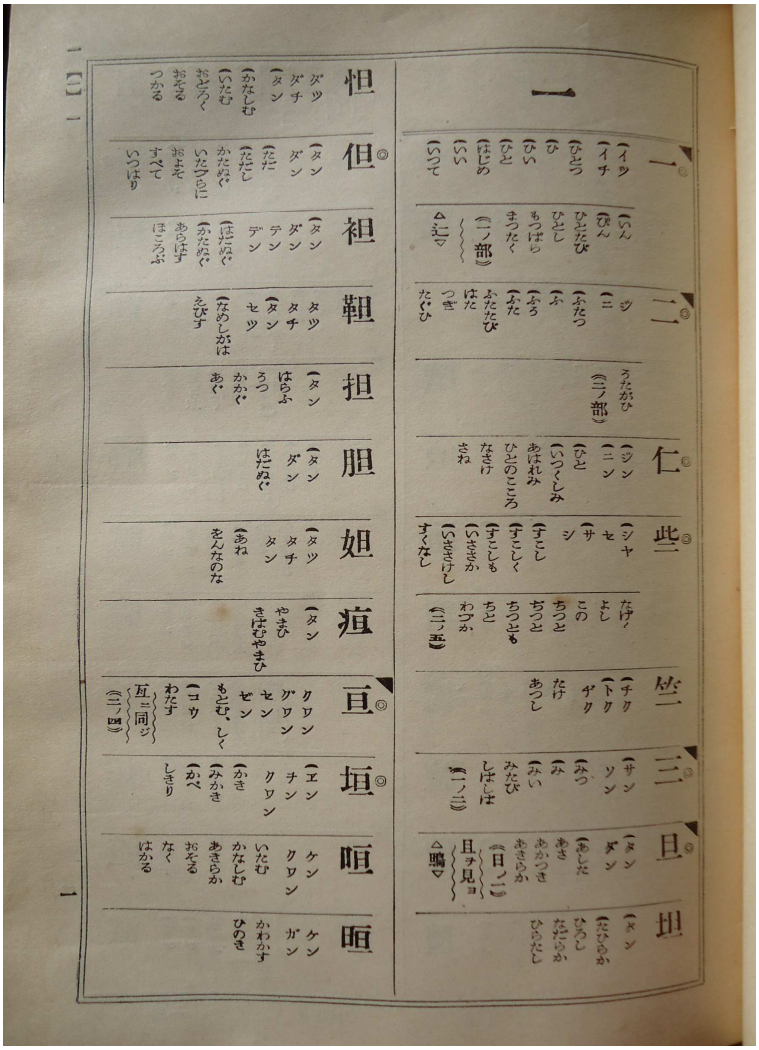


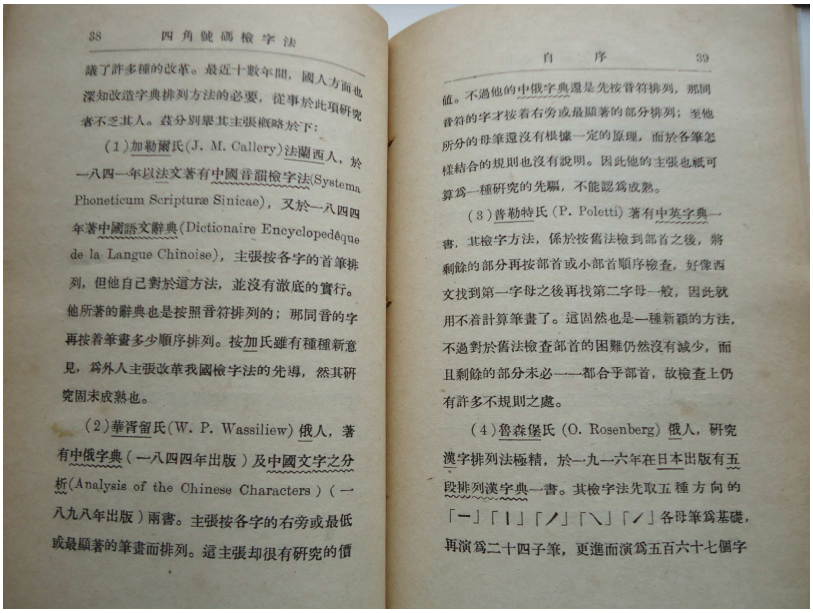
Figure 8: Examples of main entries in Rosenberg's dictionary

In the 19th century Russian researchers focused on the form of characters and, adhering to the principle of coherent classification, proposed a novel character sequencing and search method. In the former Soviet Union, this "Russian graphic system" was the basis for the Большой китайско-русский словарь (*Bol´šoj kitajsko-russkij slovar´* - "*The Great Chinese-Russian Dictionary*", Panasyuk & Suhanov, 1983). Later, the "Russian graphic system" for arranging and searching characters according to their stroke characteristics was also influential in the creation of the "Four corner method" (四角号碼 *Sì jiǎo hào mǎ*, see figures 9 and 10), developed in China during the 1920s. (Wáng, 1934, p. 38).

5.2 The Four corner method

The Four corner method (Chinese 四角號碼 *sì jiǎo hàomǎ*, Japanese 四角号碼 *skikakugōma*) is one of the Chinese character search systems and in the context of Japanese language means the "code based on four corners". It was developed in China by Wáng Yún Wǔ, who published his 號碼檢字法 (*Hàomǎ jiǎnzì fǎ - "Code based character search method"*) in 1925, 四角號碼檢字法 (*Sì jiǎo hàomǎ jiǎnzì fǎ - "Four corner code based character search method"*) in 1926, and finally 四角號碼檢字法· 附檢字表 (*Sì jiǎo hàomǎ jiǎnzì fǎ: fu jiǎnzì fǎ biao - "Four corner code based character search method: with a character look-up table appendix"*) in 1934. Just like the "Russian graphical system", the Four corner method does not depend on the radical, stroke number, stroke order, reading or meaning of a character, but rather allows for character look-up by means of a code which is based on the shape of the strokes in the four corners of a character. The stroke shapes in each of the four corners are assigned numbers from 0 to 9, and in order to distinguish between those Chinese characters which happen to end up with the same quadruplet of assigned digits, an extra "corner", named 附角 (*fujiǎo*) is additionally assigned. Each Chinese character can thus be uniquely coded and ordered by a five digit number. In order to use this Chinese character index, it is not necessary to have any knowledge of traditional search systems based on radicals, stroke number or stroke order.

Figure 9: Front page of 四角號碼檢字法 附檢字表
(Sì jiǎo hàomǎ jiǎnzì fǎ: fu jiǎnzì fǎ biao - "Four corner code based character search method: with a character look-up table appendix") (Wang, 1934)



38 四角號碼檢字法

議了許多種的改革。最近十數年間，國人方面也深知改造字典排列方法的必要，從事於此項研究者不乏其人。茲分別舉其主張概略於下：

（1）加勒爾氏(J. M. Callery)法蘭西人，於一八四一年以法文著有中國音韻檢字法(Systema Phoneticum Scripturæ Sinicae)，又於一八四四年著中國語文辭典(Dictionaire Encyclopedêque de la Langue Chinoise)，主張按各字的首筆排列，但他自己對於這方法，並沒有澈底的實行。他所著的辭典也是按照音符排列的；那同音的字再按着筆畫多少順序排列。按加氏雖有種種新意見，爲外人主張改革我國檢字法的先導，然其研究固未成熟也。

（2）華肯留氏(W. P. Wassiliew)俄人，著有中俄字典（一八四四年出版）及中國文字之分析(Analysis of the Chinese Characters)（一八九八年出版）兩書。主張按各字的右旁或最低或最顯著的筆畫而排列。這主張却很有研究的價

自 序 39

值。不過他的中俄字典還是先按音符排列，那同音符的字才按着右旁或最顯著的部分排列；至他所分的母筆還沒有根據一定的原理，而於各筆怎樣結合的規則也沒有說明。因此他的主張也祇可算爲一種研究的先驅，不能認爲成熟。

（3）普勒特氏(P. Poletti)著有中英字典一書，其檢字方法，係於按舊法檢到部首之後，將剩餘的部分再按部首或小部首順序檢查，好像西文找到第一字母之後再找第二字母一般，因此就用不着計算筆畫了。這固然也是一種新穎的方法，不過對於舊法檢查部首的困難仍然沒有減少，而且剩餘的部分未必一一都合乎部首，故檢查上仍有許多不規則之處。

（4）魯森堡氏(O. Rosenberg)俄人，研究漢字排列法極精，於一九一六年在日本出版有五段排列漢字典一書。其檢字法先取五種方向的「一」「丨」「丿」「㇏」「㇀」各母筆爲基礎，再演爲二十四子筆，更進而演爲五百六十七個字

Figure 10: Introduction to 四角號碼檢字法 附檢字表
Sì jiǎo hàomǎ jiǎnzì fǎ: fu jiǎnzì fǎ biao - “Four corner code based character search method: with a character look-up table appendix” (Wang, 1934, p. 38)

To give an example, the character 法 is assigned the code 34131, by assigning these numbers to the strokes in each corner, in the following sequence:

3 4
法 1
1 3

The Four corner method was adopted in the compilation of 大漢和辞典 (*Dai Kan-Wa Jiten*, Morohashi, 1960), *The Great Chinese Character - Japanese Dictionary* in 13 volumes published in Japan. The coding rules used in this dictionary are described in Morohashi (1984, p. 1038).

5.3 Katakana shape based classification

The Katakana shape based classification system (カタカナ字形分類索引 - *katakana jikei bunrui sakuin*, Kanō, 1998, p. 1007) sorts all the 1945 *jōyō kanji* according to the similarity of their component strokes to Japanese kana syllabary character shapes, with Chinese characters accordingly arranged as kana in the “a, i, u, e, o” order. Chinese characters are listed under the part that shares the same shape with some katakana character, as can be seen in table 2. The position of the katakana-like shape within each character is not questioned.

Table 2: Examples of Chinese characters listed under each katakana character

katakana form type	examples of corresponding Chinese characters
ア	了, 子, 孔, 好, …
イ	仙, 代, 付, 休, …
エ	工, 功, 式, 江, …
オ	才, 材, 財, 閉, …

5.4 Initial stroke pattern index

The Initial stroke pattern index (書き出しパターン索引 *kakidashi pataan sakuin*, Kanō, 1998, p. 1020) shares similarities with the aforementioned Five step arrangement Chinese character table developed by Rosenberg, but while Rosenberg operates with the strokes written at the very end, the Initial stroke pattern index deals with the strokes that are written first. The Initial stroke pattern index defines the six initial stroke patterns given in table 3.

Table 3: Initial stroke patterns in Kanō's Initial stroke pattern index (1998)

1	2	3	4	5	6
一	丨	ノ	丶	フ	レ

Characters with the same initial stroke pattern are ordered by their number of strokes. For example, characters belonging to pattern 1 (一) appear in the following order, according to their number of strokes 一, 二,丁, 三, 天, … .

5.5 Stroke order index

The Stroke order index (筆順索引 *hitsujun sakuin*) was implemented by Wakao and Hattori (1989) in their dictionary for the decipherment of cursive style characters (くずし解読字典 *Kuzusi kaidoku jiten*). Characters in this dictionary are ordered according to their stroke order and stroke direction. Brush movement is represented by arrows pointing to eight directions, with each direction being assigned a number (code) ranging from 0 to 7 (Wakao & Hattori, 1989, p. 469), as shown in table 4.

Table 4: Brush stroke direction patterns in Wakao & Hattori's Stroke order index (1989)

0	1	2	3	4	5	6	7
↑	↗	→	↘	↓	↙	←	↖

Each character is assigned a code, based on the direction of the first four strokes (起筆 *kihitsu* "first stroke", 第二筆 *dainihitsu* "second stroke", 第三筆 *daisanhitsu* "third stroke" and 第四筆 *daiyonhitsu* "fourth stroke"). For example, the first strokes of the character 仙 are written in the following directions in the cursive style: ↙ (5), ↗ (1), ↓ (4). Here the second stroke, ↗ (1), not visible in the printed form of this character, is included in cursive stroke counting because the brush is brought back to its starting point after the first stroke (Wakao & Hattori, 1989, p. 466). This coding system could easily be applied to standard character forms in general character dictionaries. In this case, the character 仙 would be coded according to its standard stroke order, ↙, ↓, ↓, ↓, resulting in the four digit code 5-4-4-4.

5.6 Key Words and Primitive Meanings Index

The Key Words and Primitive Meanings Index (Heisig, 2001, p. 506) assigns a unique meaning or interpretation to each character or character component part. English words representing these meanings are ordered alphabetically, making it possible to look up any character according to these English translations of assigned meanings. In order to use this index, the user must first learn the assigned meanings of each component part.

5.7 System of Kanji Indexing by Patterns (SKIP)

Halpern (1988/1990, 1999), developing his System of Kanji Indexing by Patterns (SKIP), assigned to each character a numeric code. In order to do this he first divided characters into four patterns, numbered from 1 to 4:

1 - ◧ Characters that can be divided into left and right parts;
2 - ⬒ Characters that can be divided into top and bottom parts;
3 - ◻ Characters that can be divided by an enclosure element;
4 - ■ Characters that cannot be classified under patterns 1, 2 or 3

These pattern numbers are used as the first digit in a code assigned to each characters. Characters of type 1 to 3 are divided into two parts. The number of strokes in each part of the character is then used as the second and third component of the character code. For example, the character 相 consists of a left and a right part and is thus categorised as type 1; the left part, 木, has 4 strokes and the right part, 目, has 5 strokes, resulting in its SKIP code 1-4-5. In the case of characters belonging to type 4, the second component of their code is the total number of strokes, while the third component is a code number, ranging from 1 to 4, assigned according to their shape. For details, readers can refer to the dictionary's detailed front matter. Further examples of SKIP codes are given in table 5 below.

Table 5: Examples of System of Kanji Indexing by Patterns (SKIP) coding

Type	Kanji	Number of strokes	SKIPcode
1 ◧	八	2	1-1-1
	相	9	1-4-5
2 ⬒	二	2	2-1-1
	父	4	2-2-2
3 ◻	山	3	3-2-1
	間	12	3-8-3
4 ■	火	4	4-4-4
	女	3	4-3-4

The dictionary contains a SKIP index constructed by ordering characters according to their SKIP codes in ascending order.

5.8 Fast Finder

Matthews (2004), in a way similar to Halpern (1988/1990) and Halpern (1999), assigns a pattern to each character, but does not create codes. Characters are divided into 8 patterns on the basis of their constituent components: left of the left-right, right of the left-right, top of the top-bottom, bottom of the top-bottom, three types of enclosures and non-divisible characters. Characters belonging to each pattern are listed together on pages beginning with the pattern itself, followed by all characters belonging to it. Lists of characters with complex shapes are further minutely subdivided and ordered according to the complexity of their shape, in ascending order of their number of strokes.

5.9 Index by Radicals

Hadamitzky and Spahn (1981) adopted an index based on traditional radicals, but reduced the number of radicals in order to simplify character search. They selected 79 of the 214 generally used radicals, which are also standardised in Unicode (Unicode, 2012), and used them to construct their Index by Radicals. Those radicals whose shapes were deemed too complex were omitted and characters traditionally assigned to them were assigned to other radicals. Some examples are given in Table 6 below. As a result of the reduced number of radicals, the number of characters listed under each radical has increased.

Table 6: Examples of radical substitution in Hadamitzky and Spahn (1981)

214 radicals in general use		79 radicals proposed by Hadamitzky and Spahn (1981)	
Radical number	**Radical**	**Radical number**	**Radical**
176	面	3s	口
177	革	3k	艹
178	韋	3d	口

5.10 Index by meaning symbols

The Index by meaning symbols (意味記号索引 *imikigō sakui*) lists characters according to 495 meaning symbols used by Kanō (1998), ordered according to their increasing number of strokes. Kanō (1998, p. 6) explains that "… We indicate the part that represents the meaning of a character as its 'meaning symbol'… Some of the 'meaning symbols' have the same shape as radicals, but are, as radicals, called differently (e.g. 水 *mizu* 'water' and 氵 *sanzui* 'three [drops of] water'). [...] It also happens sometimes that a character does not contain a 'meaning symbol'. In such cases we have shown the original character, on which the current character is based." For example, the radical of 斎 (nowadays usually substituted by 斉) is 斉, and its 'meaning symbol' is 示 (the shape of an altar) (cf. Kanō, 1998, p. 952).

5.11 Index by character shapes

The Index by character shapes (字形索引 *jikei sakuin*) is implemented in a text book including 512 Chinese characters (Banno, Ikeda, Shinagawa, Tajima & Tokashiki, 2009) and is similar to a radical index. However, the 215 "character shapes" (*jikei*) used include both radicals and other shapes which are traditionally not considered as radicals. The index lists "character shapes" ordered by their number of strokes, each followed by characters containing it, with their assigned numbers corresponding to the order in which they are introduced within the textbook.

5.12 Index by principal semantic determiners

The Index by principal semantic determiners (意符 *ifu*) developed by Shiraishi (1971/1978) is similar to a radical index, but instead of radicals relies on 243 characters representing principal semantic determiners. These determiners also include radicals, characters and character constituent elements that are not radicals. Principal semantic determiners are ordered by stroke number.

5.13 Final considerations

The above survey revealed a great variety of different types of character indexes. Among those which are based on character constituent elements and strokes, there are also indexes which assign numerical codes to characters. In order to compare and evaluate these indexes, we introduce the notion of "selectivity" (選択性 *sentakusei*). A comparison and evaluation of indexes based on selectivity is presented in the next section.

6. Evaluation and comparison of existing character indexes

6.1 Shared characteristics of existing character indexes

A point shared by all aforementioned character indexes is that only one character element or property is selected as the basis upon which the index is built. Elements or properties employed for this purpose are the number of strokes, radicals, initial stroke pattern etc. Considering the necessity to evaluate and comparatively analyse each of these indexes, we introduced the notion of selectivity of character indexes and used it to compare and evaluate existing indexes.

6.2 Definition of the "coefficient of selectivity"

To compare and evaluate the efficiency of character indexes, we will use a notion used to express computer processing efficiency, i.e., "selectivity" (Vorobeva, 2009, p. 72). The "selectivity" of an index has previously been defined as follows.

> "The ratio of the number of distinct values in the indexed column / columns to the number of records in the table represents the selectivity of an index.
>
> Example with good selectivity: a table having 100,000 records where one of its indexed columns has 88,000 distinct values, then the selectivity of this index is 88,000 / 100,000 = 0.88
>
> Example with bad selectivity: if an index on a table of 100,000 records has only 500 distinct values, then the index's selectivity is 500 / 100000 = 0.005 and in this case a query which uses the limitation of such an index will return 100000 / 500 = 200 records for each distinct value." (Akadia, 2008)

Based on the notion of selectivity, the "Coefficient of Selectivity" (hereafter CS) as a measure of index efficiency is defined as:

$$CS=V/N\times 100\%$$

where V stands for the number of distinct values in the indexed column(s) and N is the total number of records in the table.

Here, in the case of an index based on character shape, N is the total number of characters in the index, and V is the number of different groups to which characters are assigned. A group is for example a group of all characters under the same radical or a group of all characters with the same number of strokes. For example, in the case of an index by stroke numbers, V is the number of groups containing characters with the same number of strokes. In the case of an index based on radicals, V is the number of different radicals. On the other hand, in a phonetic based *on-kun* (loan and native reading) index, N is the total number of all loan (*onyomi*) and native (*kunyomi*) readings associated with all characters covered by the index, while V is the number of all distinct loan (*onyomi*) and native (*kunyomi*) readings.

To give an example of how to compute the coefficient of selectivity, let us use the old version of the *jōyō kanji* list (not the new *shin jōyō kanji*). The 1945 characters in the *jōyō kanji* list can be divided into 23 groups according to their number of strokes, or 201 groups of characters with distinct radicals (only 201 radicals are used in the 1945 jōyō kanji). Thus, for example, for the stroke number index and radical index we can compute the CS as follows:

stroke number index: V=23, N=1945, CS=23/1945×100％=1.2％

radical index: V=201, N=1945, and CS＝201/1945×100％=10.3％

From this we can conclude that the radical index is about 10 times more efficient than the stroke number based index. In the following subsections we are going to compare and evaluate the individual indexes according to this notion of “selectivity”.

6.3 Evaluation of efficiency of existing character indexes

In Vorobeva (2009), ten types of existing character indexes were compared, while in the present paper the object of comparison are 15 indexes. Existing indexes were compared on the basis of the CS (Table 7). In the table, comparison of indexes based on the character shape, character readings and character meaning is given.

It is clear from the above analysis that the CS of character shape based indexes varies between 1.2% and 25.4%, and is thus generally low. The possible reason for this is that such indexes are generally based only on one character property or element. For example, for each character, the radical index relies only on one element - the radical, the stroke number index relies only on one property - the stroke number, the initial stroke index relies only on the initial stroke, etc. Groups thus defined, i.e. groups of characters with the same radical, the same number of strokes or the same type of initial stroke, each contain a large number of different characters. On the other hand, the CS of indexes based on character reading vary in the range of 27.6~40.6％, and are more efficient if compared to character shape based indexes. However, in order to use these indexes, one has to know the readings in advance.

Table 7: Coefficients of selectivity of various character indexes

Index type	CS (%)
Indexes based on character form	
Stroke number index (Henshall, 1988)	1.2
Index by katakana shapes (Kanō, 1998)	2.6
Radical index (Hadamitzky & Spahn, 1981)	4.1
Initial stroke pattern index (Kanō, 1998)	6.1
Five step arrangement kanji table (Rosenberg, 1916)	7.1
Four corner method (Morohashi, 1984)	10.2
Radical index (Henshall, 1988)	10.3
Stroke order index (Wakao & Hattori, 1989)	10.7
Index by principal semantic determiners (Shiraishi, 1978)	12.4
Fast Finder (Matthews, 2004)	14.1
SKIP (Halpern, 1988)	15.4
Index by meaning symbols (Kanō, 1998)	25.4
Indexes based on character reading	
Index by principal phonetic determiners (Shiraishi, 1978)	27.6
On-kun reading index (Henshall, 1988)	40.6
Index based on character meaning	
Key Words Index and Primitive Meanings (Heisig, 2001)	100.0

Indexes based on character meaning, such as the Key Words Index and Primitive Meanings (Heisig, 2001) have a CS reaching 100%, but require the user to know in advance the meaning of all characters.

Based on these findings it can be said that indexes which rely on the coded total shape of a character, such as the two indexes above, are much more efficient then indexes relying on a single element or property. It can therefore be concluded that such an index is needed for efficient character searching. Thus, in the following section, a new efficient index will be proposed and developed. For this purpose, character were structurally decomposed and coded.

7. Structural decomposition and coding of characters

Three coding systems based on the structural decomposition of characters and three character databases built on the basis of these systems are introduced by Vorobeva (2011). In the present paper, we will discuss four systems of character coding. All characters contained in the *jōyō kanji* list, and also those added in the *shin jōyō kanji* list were encoded, and four databases were built on the basis of the following codes:

(1) an alphabet code
(2) a symbol code (containing roman letters and digits),
(3) a semantic code (expressed by words),
(4) a radical and stroke codes.

Structural decomposition of characters and semantic analysis of their constituent elements stimulate a deeper understanding of character meaning. Structural decomposition of characters can be achieved at two levels, i.e., decomposition into strokes (書記素 *shokiso*) and decomposition into constituent elements (the smallest meaningful constituent elements of Chinese characters, 構成要素 *kōseiyōso*).

Decomposition into strokes follows the stroke order, as in the following example:

女 → {く, 丿, 一}

Decomposition into constituent elements also follows the stroke order, but to obtain the smallest meaningful constituent elements, as in the following example:

露 → {雨, 足, 夂, 口}

After having defined the alphabet and number codes of strokes and constituent elements of each character, and the alphabet and number codes for each whole character, the complete *shin jōyō kanji* list was encoded.

7.1 Type of strokes and their encoding

7.1.1 The alphabet code of strokes

According to Zadoenko and Khuan (1993), Chinese characters used in China can be decomposed into strokes belonging to 24 different types. Fazzioli (1987) also uses almost the same strokes. In an analysis of the shapes of all characters in the *shin jōyō kanji* list, we found that the 24 types of strokes proposed by Zadoenko and Khuan (1993) are necessary and sufficient. We therefore coded these 24 shapes into Latin alphabet letters from A to Z based on the similarity of their shapes, so that for each stroke a corresponding letter could be guessed on the basis of its shape (cf. Table 8).

A	B	C	D	E	F
G	H	J	K	L	M
N	O	P	Q	R	S
T	U	V	W	Y	Z

Table 8: The 24 character strokes and their alphabet codes

7.1.2 The alphabet coding of characters and its use

An alphabet code was obtained for each character by following its stroke order and transforming each stroke into its corresponding alphabet code as given in table 8. Thus, the stroke order for each character could also be expressed by an alphabet code, as in the following examples:

三（AAA） 川（PBB） 玉（ABAAQ） 女（KPA） 小（JLQ）

All characters in the *shin jōyō kanji* list were encoded according to this procedure, and the alphabetic order of the code thus obtained was used to arrange the entries in a new character dictionary index. In this way a new index of character entries could be developed. An excerpt of the index is given in section 8, table 12.

7.2 Types of constituent elements and their encoding

7.2.1 Types of constituent elements

There are two types constituent elements which can form characters: elements which are radicals and elements which are not traditionally considered radicals but are patterns which correspond to radicals in some aspects. We name the latter “graphemes”. Constituents combine in complex ways to form individual characters.

7.2.2 Types and coding of radicals

Traditionally, 214 types of graphic patterns have been considered as radicals, as mentioned in the previous sections. Unicode 6.1.0 (Unicode, 2012) includes a table of these 214 radicals, and each radical is assigned a unique number. Analysing both the *jōyō kanji* list and the *shin jōyō kanji* list, we found that of these traditional 214 radicals, 201 types could be found as constituent elements of characters in the *jōyō kanji* list and 202 types as constituent elements of characters in the *shin jōyō kanji* list. In the present study, we assigned three different codes to each radical:

(1) an alphabetic code composed of the alphabetic codes of all strokes which make up the radical;

(2) a symbol code corresponding to the traditional numbering of radicals from 1 to 214;

(3) a semantic code consisting of a word expressing the principal meaning of each radical.

Examples of these codes are given in table 9.

Table 9: Examples of coded radicals

Radical	Alphabet code	Symbol code (number)	Semantic code (Principal meaning)
一	A	1	one
丨	B	2	stick
人	PO	9	man
山	BEB	46	mountain
馬	BABAAGLQQQ	187	horse

7.2.3 Types and coding of graphemes

Vorobeva (2011, p. 11) structurally decomposed each character from the *shin jōyō kanji* list and extracted all graphemes included in them, resulting in 161 different types of graphemes. This extraction was based on an intuitive notion of what would constitute a grapheme. In the present study, we first defined some rules to determine what part of the character can be extracted as a grapheme. We then structurally decomposed each character in the *shin jōyō kanji* list according to these rules, extracted their graphemes and found 220 different types of graphemes. Combining this list with the 202 traditional radicals found in the previous analysis, we found that the 2136 characters in the *shin jōyō kanji* list are composed of 422 different constituent elements.

The rules used to structurally decompose characters and extract graphemes are based on Stalph (1989, p. 69) and are given below.

1. Graphemes are limited to characters or character constituent elements included either in Unicode 6.1.0 (Unicode, 2012) or the "*Mojikyō tankanji*" list (Mojikyō Kenkyūkai, 2002). Unicode 6.1.0 contains 74,617 characters listed as CJK Unified Ideographs, and the "*Mojikyō tankanji*" list contains 110,000 characters. The reason for using these lists is that when characters are decomposed, the decomposition may result in shapes that do not appear in the *shin jōyō kanji* list.

2. When the decomposition of characters or constituent elements into smaller elements reaches a point where a stroke does not constitute an independent character, the decomposition is stopped and such a shape is considered a grapheme. For example: decomposing 及 further would give two elements: 乃 and 𠂉, but since 𠂉 is not an independent element, the decomposition is stopped and 及 is considered itself a grapheme. In shorthand notation: 及 ≠ 乃 + 𠂉

Attention is necessary in the case of 一 and 乙. Here they are considered as characters and not as strokes.

3. When the decomposition of a character or a constituent element reaches an element which by itself is not a radical but is a constant shape always used in

combination with a radical or other graphemes, this shape is considered a grapheme, without further decomposition. For example, the constituent element 辟 appears in characters such as 避, 壁, 癖, and 璧. It contains the element 尸 which by itself is not a radical and which regularly accompanies the radical 辛. This constituent element 辟 is not further decomposed, i.e. 辟 ≠ 尸 + 辛, and 辟 is considered as a grapheme.

4. When decomposing a character or a constituent element, if it is necessary to cut one stroke to obtain two independent shapes and if the total number of resulting strokes is then higher than in the original character or constituent element, then such character or constituent element is considered as a grapheme and is not decomposed further. For example:

出 ≠ 山 + 山, 重 ≠ 千 + 里.

Thus, 出 and 重 are considered as graphemes.

5. When decomposing a character or a constituent element, if it is necessary to insert a stroke into a decomposed element to make it complete and if the total number of resulting strokes of the two parts exceeds the number of strokes in the original character, then such a character or constituent element is considered a grapheme and not decomposed. For example: 雀 ≠ 少 + 隹 and thus 雀 is considered a grapheme.

6. When decomposing a character or a constituent element, if it is necessary to split two crossing strokes to arrive at a grapheme or a radical, such a character or constituent element is considered a grapheme. For example: 必 ≠ 心 + ノ, and thus 必 is a grapheme.

7. A character or constituent element which is deemed impossible to further decompose is considered a grapheme. For example: 奈, 喬, 兼, etc, are considered graphemes.

As the next step, all graphemes were assigned three types of codes according to the same procedure used for radicals. Details about encoding are given in Vorobeva (2011, p. 21). Examples of codes for graphemes which are component parts of characters in the *shin jōyō kanji* list are given in table 10 below.

Table 10: Examples of coding of graphemes included in characters in the *shin jōyō kanji* list

Grapheme	Alphabet code	Symbol code	Semantic code
丁	AJ	2AJ	street
丂	AN	2AN	snare
マ	YQ	2YQ	chop-seal
人	POA	3POA	meeting

7.2.4 Symbol codes, semantic code, radical and stroke code, and their use

Symbol codes and semantic codes of character are described in Vorobeva (2007, p. 22). The symbol code of each character is obtained by listing the symbol codes of each constituent element (cf. 7.2.2), following the stroke order.

The semantic code of a character is defined as the semantic code of its first two constituent elements, following the stroke order. Only the first two elements are used in order to avoid long codes.

Further, codes based on radicals and strokes are obtained by listing the numbers of elements which correspond to traditional radicals and the symbol codes (roman letters) assigned to each of the remaining strokes, following standard stroke order. Letters indicating strokes and numbers indicating radical shapes are divided by slashes, as in the following examples.

決 → {氵, 冂, 大} → 85 / H / 37,

今 → {亼, 一, フ} → 9 / 1 / Y。

Numbers and letters used in the above code are, for radicals, standard radical numbers from Unicode 6.1.0 (Unicode, 2012), and for strokes, the alphabet codes assigned to basic strokes (cf. table 8).

Following the above procedure, all characters in the *shin jōyō kanji* list were encoded and a database was constructed to include the alphabet codes, symbol codes, semantic codes, as well as radical and stroke codes. Examples of such coding are given in table 11 below.

Table 11: Examples of alphabet, symbol, semantic and radical-stroke character codes

character	Alphabet code	Symbol code	Semantic code	Radical and stroke code
九	PR	2PR	nine	4/5
逸	PYBHBAPCQMO	8PYB/162	escape/road	18/30/2/10/162
新	SAQLAABPOPPAB	117/75/69	stand/tree	117/75/69

8. Towards a new type of character index based on character coding

8.1 Construction and use of a new type of character index

Starting from the premise that it is necessary to develop a character index which would make search in character dictionaries more efficient and which would be suitable for learners not familiar with Chinese character writing, we developed a character code based on appropriate representations of character shapes (cf. Vorobeva

2007, 2009, p. 72). In order to develop such a new type of character index, we sorted the alphabet codes, symbol codes, semantic codes, and radical/stroke codes for all characters in the *shin jōyō kanji* list in standard dictionary order (alphabetical and numerical order). Searching for characters in such indexes should require the same amount of labour as searching for words in alphabetically ordered dictionaries to which learners from non-kanji background are accustomed.

A symbol code index and a semantic code index were developed and implemented in two character textbooks including 518 characters, *Kanji monogatari I* [漢字物語Ⅰ] (Vorobeva, 2007) and *Kanji monogatari II* [漢字物語Ⅱ] (Vorobev & Vorobeva, 2007).

Further, an alphabet code index, a symbol code index and a semantic code index were developed for the 1945 characters in the *jōyō kanji* list, selectivity coefficients were computed and compared with existing character indexes (Vorobeva, 2009).

In 2010, after the new *shin jōyō kanji* list with 2136 characters was approved, an alphabet code index, a symbol code index and a semantic code index were compiled for the new list, and a new, easier to use radical-and-stroke code index was also developed. The next sections introduce each of these new indexes.

8.2 Alphabet code index

The first part of the alphabet code index is shown in Table 12. In order to be able to use this index, one needs to memorise the 24 types of strokes and the rules governing stroke order of characters.

Table 12: First part of the alphabet code index for all characters in the *jōyō kanji* list

Alphabet coding of character	character
A	一
AA	二
AAA	三
AAABPQAAPB	耕
AAABPQPAAC	耗

Beginning learners are not yet accustomed to the constituent elements of characters. The alphabet code index is therefore expected to be easier to understand and use.

8.3 Symbol code index

Most characters are composite characters, composed of multiple constituent elements. We coded all characters in our textbook using radical and grapheme codes, compiled a database of characters with their respective radical and grapheme codes, sorted the database according to the code column, in standard dictionary ascending order (i.e. alphabetic and numerical order), and thus obtained a symbol code index. Users can thus search for characters in this index using codes based on character shape.

For example, the symbol code for the character 親 is 117/75/147. The numbers in the code are the numbers of the radical shapes into which the character can be decomposed, i.e. 立 (117), 木 (75), and 見 (147). To use the symbol code index, users refer to the table of radical numbers and grapheme codes. After some time, users generally end up remembering the numbers and codes by using them.

An excerpt from the symbol code index in *Kanji monogatari I* [漢字物語Ⅰ] (Vorobeva, 2007) and *Kanji monogatari II* [漢字物語Ⅱ] (Vorobev & Vorobeva, 2007) is shown in table 13.

Table 13: First part of the symbol code index in the textbooks *Kanji monogatari I and II*

character symbol code	**character**	**character number in *Kanji monogatari I,II***
1	一	11
1/106	百	21
1/119	来	51
1/13/46	両	238
1/132/34	夏	189

8.4 Semantic code index

Part of the semantic code index from the textbooks *Kanji monogatari I* and *II* is given in table 14. When using the semantic code index, users refer to the list of words representing the meaning labels of character constituent elements. The semantic code is obtained by listing the meaning labels of the first two constituent elements, following standard stroke order. For example, the semantic code for the character 新 is “stand/tree”, since the first two constituent elements and their corresponding meaning labels are 立 (stand), and 木 (tree). In order for the semantic code not to be too long, only the meaning labels of the first two constituent elements are used.

Table 14: First part of the semantic code index in the textbooks *Kanji monogatari I and II*

character semantic code (Russian /English)	**character**	**character number in *Kanji monogatari I,II***
Азия/сердце (Asia / heart)	悪	140
бамбук/встреча (bamboo/ meeting)	答	447
бамбук/дерево (bamboo/tree)	箱	449

8.5 Radical and stroke index

The special characteristic of this index is that users only need to remember the radicals and the basic strokes in order to use it. The first part of the radical and stroke code index is shown in Table 15.

Table 15: First part of the radical and stroke index

Radical and stroke code	**character**
1	一
1/102/17	画
1/106	百
1/132/34	夏
1/25	下
1/30/6/1/30/6/76	歌

The nine types of strokes (A, B, C, F, J, P, Q, R, W), given in Table 8 are at the same time also constituent elements appearing in the table of radicals. In the compilation of the radical and stroke based index, these strokes were treated as radicals. By analysing all characters in the *shin jōyō kanji* list, we found that more than 90% of these characters are entirely composed of elements which can be found in the traditional list of radicals. Characters that include other strokes (i.e. D, E, G, H, K, L, M, N, O, R, S, T, V, Y, Z in table 8) are relatively few, less than 10% of all characters in the *shin jōyō kanji* list. The radical and stroke code, which is mainly based on radicals, is therefore presumably easier to acquire and use than the symbol code index and the semantic code index.

Table 16 shows the frequency of use of stroke codes in the radical and stroke code index for all characters in the *shin jōyō kanji* list.

Table 16: Frequency of use of stroke codes in the radical and stroke code index for characters in the *shin jōyō kanji* list

Stroke code	**D**	**E**	**G**	**H**	**K**	**L**	**M**	**N**	**O**	**R**	**S**	**T**	**V**	**Y**	**Z**
Frequency of use	0	25	20	19	5	42	4	16	51	2	0	9	19	0	4

8.6 Comparison and evaluation of the new type of character indexes

If the new character indexes introduced in the preceding sections are learned and used, the workload necessary for search is comparable to the search in alphabetic dictionaries, resulting in more efficient character search. At the same time it can be expected that the learners gain a better insight into the structure of characters and are thus freed from rote memorisation. Table 17 gives the coefficients of selectivity (CS) of these new indexes, computed for the 1945 characters the *jōyō kanji* list. CS for each index is obtained by dividing the number of distinct characters under each code by the total number of characters in the index, and multiplying it by 100, as explained in section 6.2.

The 1945 characters in the *jōyō kanji* list were considered instead of the 2136 characters from the *shin jōyō kanji* list, in order to facilitate a comparison with existing indexes.

Table 17: Coefficients of selectivity (CS) for the new types of indexes

New type of index	**Coefficient of selectivity (%)**
Semantic code index	64.1
Alphabet code index	98.4
Symbol code index	99.4
Radical and stroke index	99.4

The results of the analysis show that the alphabet code index, the symbol code index and the radical and stroke code index all have a CS close to 100%, implying better ease of search compared with existing indexes based on character shape. In other words, in the new indexes, different characters sharing the same code are few, and many characters have a unique code. For example, in the radical and stroke code index, there are only 11 characters sharing their code with other characters (cf. table 18). CS values for the radical and stroke code index for characters in the *jōyō kanji* list and *shin jōyō kanji* list are:

jōyō kanji list: CS = (2136-11)/2136*100 = 99.5%
shin jōyō kanji list :CS = (1945-11)/1945*100 = 99.4%

Table 18: Characters from the *shin jōyō kanji* list sharing the same code in the radical and stroke code index

radical and stroke code	character 1	character 2	character 3	character 4
96	王	玉		
102	田	申	由	甲
57/2	引	弔		
9/7/28	会	伝		
30/154	員	唄		
120/102	細	紳		
83	氏	民		
1/75	末	未		
64/102	押	抽		

The CS value for the semantic code index is 64.1%, lower than the CS for the alphabetic code index, the symbol code index and the radical and stroke index. This is because the semantic code only includes the codes of the first two elements in a character, resulting in relatively many sets of characters with the same code.

In order to use character coding systems, some effort is required. For the new coding systems proposed above, it is especially important to accurately master the rules governing stroke order and the decomposition of characters into constituent elements and strokes.

The alphabet code index is probably the easiest to learn among the new indexes proposed, since users only need to memorise 24 types of basic strokes and their corresponding alphabetic codes. This index can therefore be used from the beginning stages of character instruction.

In order to use the radical and stroke index, learners need to memorise radicals, their numeric codes, and 24 basic strokes with their alphabetic codes.

In order to use the symbol index and the semantic code index, learners need to memorise radicals with their respective numerical codes, and graphemes with their alphabetic codes.

However, the symbol code index, the semantic code index, and the radical and stroke index have two merits if compared with the alphabetic code.

(1) The code is short. For example, the alphabetic code for the character 高 is SABHABGBHA, while its symbol code and its radical and stroke code (identical in both cases) is 189.

(2) In order to use the radical and stroke code, users need to learn how to structurally decompose characters and to learn about their constituent parts, which enhances comprehension and memorisation of characters.

8.7 Uses of the new index types

At the beginning stage of learning to read and write Chinese characters, the alphabetic index is probably easier to use than other indexes, since learners do not yet know the constituent elements which make up characters. However, as learning progresses, characters to be learned become more complex, being made up of more strokes, and the alphabetic code becomes longer. At the same time, learners progressively learn to recognise different constituent parts within complex characters, and are able to use other indexes, choosing the one that best suits their learning style, be it the symbol code index, the semantic code index, or the radical and stroke code index. At this point, they usually start using mainly one of these indexes.

The radical and stroke index is probably easier to learn and use than the symbol code and the semantic code index, but the latter two are probably more useful for deepening learners' understanding of character structural composition.

We used the symbol code index and the semantic code index in two textbooks we developed, *Kanji monogatari I* [漢字物語Ⅰ] (Vorobeva, 2007) and *Kanji monogatari II* [漢字物語Ⅱ] (Vorobev & Vorobeva, 2007). In the following paragraphs we explain how the use of these indexes was introduced to our learners.

In order to look up an unknown character in the textbook, learners firstly write the first two constituent elements of the character and convert them into their respective codes. At first, when they are yet not accustomed to characters, they actually write down the first elements on paper, later they learn to just imagine writing the character and convert the first two elements into their codes. They then look through either the symbol code index or the semantic code index, where codes are arranged in alpha-numeric order, and look up the character as they would in a dictionary of alphabetically arranged words, until they find the number of the character in the index. Using this number, they can then find the character inside the textbook and read the textbook explanation about the character they were searching for. Searching in these two types of indexes is equivalent to searching through an alphabetically ordered dictionary. The procedure (algorithm) used when searching the aphanumeric symbol code index is described by Vorobeva (2007, p. 25). Through use, learners get progressively accustomed to this way of looking up characters, and after some practice are able to find a character in a few seconds.

9. Selectivity of character indexes and learning burden

In the present paper, we defined the Coefficient of Selectivity as an index of efficiency for character indexes, and compared existing indexes with our newly developed index types on the basis of this coefficient. However, there is one more factor to be considered when considering the efficiency of indexes, i.e. the learning burden required from users, the special knowledge they need to acquire in order to be

able to actually use each of these indexes, such as character radicals, types of strokes and their counting, readings and meanings associated with each character, their coding etc. When discussing the efficiency of indexes, such learning burden must also be considered, as discussed in Vorobeva (2011, p. 24).

The index by total number of strokes is probably the easiest to master among the indexes presented in the previous sections. However, given its CS of 1.2%, it is clearly not an efficient index. On the other hand, the Key Words and Primitive Meanings Index (Heisig, 2001), with a CS of almost 100%, allows for very efficient search, but in order to use it, learners need to memorise approximately 2000 semantic keywords. We can therefore conclude that in future research it will be necessary to analyse the learning burden of character indexes, and further define a comprehensive index of efficiency which would reflect both selectivity and learning burden of character indexes.

10. Conclusion and further work

In the present paper we discussed the difficulties faced by learners of Japanese not familiar with Chinese characters when they use character dictionaries, presented 15 types of existing character indexes and described their characteristics. We pointed out that for users who need to look up characters in a dictionary, in addition to the generally used radical index, stroke number index and readings index, many other different types of character indexes have been developed and used, including the five step arrangement kanji table (Rosenberg, 1916), the four corner method (Wang, 1925), the phonetic key index (Shiraishi, 1971/1978), the index of katakana shapes, the index of initial stroke patterns and the index of meaning symbols (Kanō, 1998), the stroke order index (Wakao & Hattori, 1989), the index of character shapes (Sakano, Ikeda, Shinagawa, Tajima, & Tokashiki, 2009), the key words and primitive meanings index (Heisig, 1977/2001), the index by radicals (Hadamitzky & Spahn, 1981), the system of kanji indexing by patterns - SKIP (Halpern, 1988), the kanji fast finder (Matthews, 2004) and others.

In order to compare the effectiveness of different character indexes, we applied the concept of selectivity, a concept used to express the efficiency of computer data processing, to character indexes, and defined the concept of coefficient of selectivity (CS) as an index of efficiency of character indexes. We then computed CS for existing character indexes and compared their efficiency. We found that indexes based on character form or structure had a low CS in the range of 1.2% to 25.4%, probably because most indexes based on character form use only one structural element or characteristic of each character for indexing, such as the radical index using only the radical part of a character, the total stroke number index using only the number of strokes, or the initial stroke pattern index using only the form of the first stroke for indexing characters.

In order to improve the efficiency of character dictionary indexes, we considered it necessary to develop a new type of index with a hight selectivity coefficient that would be appropriate for the habits of learners not familiar with Chinese character writing. We therefore developed an alphabetic and a symbol code index, based on a code which accurately represents the form of Chinese characters, a semantic code index and a radical and stroke code index. In this paper, we described the use of these indexes, and comparatively evaluated their efficiency. We found that the alphabetic code index, the symbol code index and the radical and stroke index have a coefficient of selectivity nearing 100%, while the semantic code index only has a coefficient of selectivity of 64.1%. The reason for such a low CS is probably that it only takes into account the first two structural elements of each character, which leads to a relatively high number of characters with the same code.

We expect that learners using the above new types of indexes should be able to look up characters in dictionaries more efficiently, deepen their understanding of character structure, and be emancipated from rote learning. We implemented these new types of indexes in two introductory textbooks including 518 characters, *Kanji monogatari I* (Vorobeva, 2007) and *Kanji monogatari II* (Vorobev & Vorobeva, 2007). We plan to include these new types of indexes (an alphabetic code index, a symbol code index, a semantic code index, and a radical and stroke index) in a new textbook with 1006 characters for the initial and intermediate level that is under development, and to empirically investigate the efficiency of these indexes by surveying how learners use them in practice.

Moreover, the usefulness of character indexes should be investigated by measuring not only their coefficient of selectivity, but also the learning burden associated with them, as suggested in Vorobeva (2011, p. 24). We are therefore planning to further investigate both factors and define a comprehensive measure of efficiency for character dictionaries.

References

Akadia. (2008). *How to measure index selectivity*. Retrieved July 27, 2012, from http://www.akadia.com/services/ora_index_selectivity.html

Banno, E. [坂野永理], Ikeda, Y. [池田庸子], Shinagawa, C. [品川恭子], Tajima, K. [田嶋香織], & Tokashiki, K. [渡嘉敷恭子]. (2009). *Kanji look and learn : 512 kanji with illustrations and mnemonic hints : imeeji de oboeru 'genki' na kanji 512 : genki plus* [Kanji look and learn : イメージで覚える「げんき」な漢字 512 : genki plus]. Tokyo: The Japan Times.

Fazzioli, E. (1987). *Chinese calligraphy: from pictograph to ideogram: the history of 214 essential Chinese/Japanese characters*. New York: Abbeville Press.

Hadamitzky, W., & Spahn, M. (1981/1997). *Kanji & kana: A handbook of the Japanese writing system*. Boston: Tuttle.

Halpern, J. (1988). *New Japanese-English character dictionary*. Tokyo: Kenkyusha.

Heisig, J. (1977). *Remembering the kanji. Vol. 1*. Tokyo: Japan Publications Trading.

Henshall, K. (1988). *A guide to remembering Japanese characters*. Boston: Tuttle.

Kanō, Y. [加納喜光] (1998). *Jōyō kanji mirakuru masutā jiten* [常用漢字ミラクルマスター辞典]. Tokyo: Shogakukan [小学館].

Kod_Rozenberga [Код_Розенберга]. (2012, July 27). Retrieved July 27, 2012 from http://www.enci.ru/Код_Розенберга

Matthews, L. (2004). *Kanji fast finder - Kanji hayabiki jiten* [漢字早引き辞典]. Boston: Tuttle.

Mojikyō Kenkyūkai [文字鏡研究会]. (2002). *Pasokon yūyū kanji jutsu: Konjaku mojikyō tettei katsuyō* [パソコン悠悠漢字術:今昔文字文字鏡徹底活用]. (3rd ed.). Tokyo: Kinokuniya shoten [紀伊國屋書店].

Morohashi, T. [諸橋轍次] (1960/1984). Dai kanwa jiten [大漢和辞典]. Tokyo: Taishūkan [大修館書店].

Panasyuk, V.A. [Панасюк В.А.], & Suhanov V.F. [Суханов В.Ф.] (1983). *Bol´šoj kitajsko-russkij slovar´ - hua e da ci dian* [Большой китайско-русский словарь 華俄大辞典]. Moscow: Nauka [Наука].

Pešurov, D. A. [Пещуров Д.А.] (1891). *Kitajsko-russkij slovar´ - Po grafičeskoj sisteme* [Китайско-русский словарь. По графической системе]. Saint Petersburg: Tipografija Imperatorskoj Akademii Nauk [СПб., Типография Императорской Академии Наук].

Rosenberg, O. [ロゼンベルグ・オ] (1916). *Godan hairetsu kanjiten* [五段排列漢字典] - *Arrangement of the Chinese characters according to an alphabetical system with Japanese dictionary of 8000 characters and list of 22000 characters.* Tokyo: Kōbunsha [興文社].

Shiraishi, M. [白石光邦]. (1971/1978). *Yōsokeiteki kanji gakushū shidōhō* [要素形的漢字学習指導法]. Tokyo: Ōfūsha [桜楓社].

Stalph, J. (1989). *Grundlagen einer Grammatik der sinojapanischen Schrift*. Wiesbaden: Otto Harrassowitz.

Unicode. (2012). *Unicode 6.1.0*. Retrieved March 15, 2012 from http://www.unicode.org/versions/Unicode6.1.0/

Vasil´ev, V. P. [Васильев В. П.] (1867). *Grafičeskaâ sistema kitajskih ieroglifov. Opyt pervogo kitajsko-russkogo slovarâ* [Графическая система китайских иероглифов. Опыт первого китайско-русского словаря] [scholarly edition published online in 2010 at http://www.ci.spbu.ru/slovar/index.html].

Vorobeva, G. [ヴォロビヨワ・ガリーナ] (2007). *Kanji monogatari I* [漢字物語Ⅰ]. (2nd ed.). Bishkek: Lakprint.

Vorobev, V. [ヴォロビヨフ・ヴィクトル], & Vorobeva, G. [ヴォロビヨワ・ガリーナ] (2007). *Kanji monogatari II* [漢字物語Ⅱ]. Bishkek: Lakprint.

Vorobeva, G. [ヴォロビヨワ・ガリーナ] (2009). Sentakusei ga takai kanji sakuin no kaihatsu [選択性が高い漢字索引の開発]. *Nihongo kyōiku hōhō kenkyū kaishi* [日本語教育方法研究会誌], *16* (1), 72-73.

Vorobeva, G. [ヴォロビヨワ・ガリーナ] (2011). Kōzō bunseki to kōdoka ni motozuku kanji jitai jōhō shori shisutemu no kaihatsu [構造分析とコード化に基づく漢字字体情報処理システムの開発]. *Nihongo kyōiku* [日本語教育], (149), 16-30.

Wakao, S.[若尾俊平], & Hattori D. [服部大超]. (1989). *Kuzushi kaidoku jiten* [くずし解読字典]. Tokyo: Kashiwa Shobō [栢書房].

Wáng, Yún Wǔ [王雲五]. (1925). *Hào mǎ jiǎn zì fǎ* [號碼檢字法]. Shanghai: Shāngwù yìnshūguǎn [商務印書館].

Wáng, Yún Wǔ [王雲五] (1926). *Sì jiǎo hào mǎ jiǎn zì fǎ* [四角號碼檢字法]. Shanghai: Shāngwù yìnshūguǎn [商務印書館].

Wáng, Yún Wǔ [王雲五]. (1934). *Sì jiǎo hào mǎ jiǎnzìfǎ·fù jiǎn zì fǎ* [四角號碼檢字法·附檢字表]. Shanghai: Shāngwù yìnshūguǎn [商務印書館].

Zadoenko T. P. [Задоенко Т. П.], & Khuan, S. [Шуин Хуан] (1993). *Osnovy kitajskogo jazyka: vvodnyj kurs - Chi ch'u Han yü* [Основы китайского языка: вводный курс - 基礎漢語]. Moscow: Nauka [Наука].