Surface or Essence: Beyond the Coded Character Set Model.

Shigeki Moro[1)]

Abstract

For almost all users, the coded character set model is the only way to use characters with their computers. Although there have been frequent arguments about the many problems of coded character sets, until now, there was almost nothing on the philosophical consideration on a character in the field of Computer science. In this paper, the similarity between the coded character set model and Aristotle's Essentialism and the consequent problems derived from it, is discussed. Then the importance of the surface of the character is pointed out using the écriture theory of Jacques Derrida. Lastly, the Chaon model of the CHISE project is introduced as one of the solutions to this problem.

Keywords: Unicode, Aristotle's Essentialism, Derrida's Theory of écriture, Chaon model

"*Depth must be hidden. Where? On the surface.*"
—Hugo von Hofmannsthal (1874-1929)

1 Introduction.

Writing, is not only considered as one of the most fundamental mediums of intellectual activities, but also a frequently used one, which is not restricted to the use of computers alone. Needless to say that the coded character set model (abbreviation being CCSM) is the most popular one to encode characters in computers and we have many coded character sets based on this model.

In recent years, the number of characters, which we can use in our computers, has increased dramatically and our former state of starvation caused by the lack of characters has been partly filled. However, when compared to the number of the increased characters, I suppose the quotient of our happiness has not increased proportionately. For example, the available standards do not fully provide the knowledge required to use a character in computers, such as, how to search a character, how to control layout, context-dependent variations and so on.

Moreover, local character sets have been developed for domestic purposes and super sets for them, such as Unicode ([24]), TRON code ([23]) and so on, have been developed based on today's global computing circumstances[1]. Although Unicode is a well-developed implementation based on this model, other local and super character code sets are still being developed, and the repertoires of the existing character sets are increasing even now. What users can only do is to choose and follow these character sets.

The main reason for this is that there are both sides: Writing is not only dependent on a context, but that it is transmitted exceeding the context (it is contrastive with oral language being indivisible from a context). For instance, an alphabet "a" is pronounced differently based on the contexts, such as "c*a*t," "c*a*ke," and so on. It's, on the other hand, used in the various Latin scripts. TRON code, in which a Chinese character 一 from JIS X 0208 and 一 from GB2312 are encoded separately in order to distinguish Japanese and Chinese[2], is a poor implementation which purports that the context-dependency is important. In other words, it aims to express contexts by code points. On the other hand, the unification policy of the Unicode Standard regards the context-independency as important. Such various situations hint that the duality of a character cannot be well expressed by CCSM.

This paper deals with two main considerations: First, we will consider the fundamental misunderstanding of a character, which CCSM holds. For this investigation, I would like to focus on Unicode since it has the well-defined design principle which is hardly discussed in other standards and to use the method of comparative philosophy in order to clarify the problems, especially with Aristotle's es-

1) Hanazono University. (s-moro@hanazono.ac.jp)

[1] GB18030 of the People's Republic of China is an interesting example since it has a hybrid nature of both local and super character sets. See [18].

[2] In addition, 一 s from KS X 1001, GT Mincho, CNS 11643, Morohashi Daikanwa and Unicode (!) are not unified in TRON code.

sentialism and Jacques Derrida's theory of écriture. In other words, the view of Unicode about a character will be located in European intellectual history.

Second, the Chaon model proposed by Dr. Tomohiko Morioka, who is the leader of the CHISE project, will be introduced as one of the solutions to this problem. The CHISE project, of which I am a developer, aims to create a new character / text processing environment, independent of any character code sets or CCSM. The Chaon model is a fundamental model for our project.

Although these subjects may not be directly related to the problems of typesetting and fonts, it is reasonable to study them since they are the subjects relevant to KAGE and Ω/CHISE system[3], which are a part of the CHISE project to develop a new font and typesetting environment.

2 Essentialism of Unicode.

2.1 Definition of *characters*.

Let us begin our analysis by looking at "10 Unicode Design Principles" of the Unicode Standard quoted below ([24], p. 14).

1. Universality[4]
2. Efficiency
3. Characters, Not Glyphs
4. Semantics
5. Plain Text
6. Logical Order
7. Unification
8. Dynamic Composition
9. Equivalent Sequences
10. Convertibility

Of course, we maniacs of character codes and typesetting, know that not all these principles are in principle or deductive, but distorted by historical, political and religious reasons and also know that these definitions are not completely put into practice. Although it is also interesting to explore the complicated historical situations of the development of Unicode, it's beyond the scope of this paper. The philosophical background of these principles will be examined.

Especially for our examination, the third, fourth, sixth and seventh principles are important. In the third principle, Unicode defines the term *character* as follows:

> The Unicode Standard draws a distinction between *characters* and *glyphs*. Characters are the abstract representations of the smallest components of written language that have semantic value. (...) Characters represented by code points. (...) The Unicode Standard deals only with character codes. ([24], p. 15)

This principle may be the most important one to investigate the philosophy of Unicode. According to this principle, there is an another existence before an actual character, i.e., a written glyph, which exists physically and a single code point can express a *character*. By this it's easy to be reminded of Plato's idealism, ātman in Indian philosophy and so forth. Therefore, a *character* is not a character but what should be called a "pre-character" or an "archi-character".

In addition, a sequence made of *characters* should be regarded as something like a "pre-text" and so on. The sixth principle is concerned with this issue:

> Unicode text is stored in *logical order* in the memory representation, roughly corresponding to the order in which text is typed in via the keyboard. In some circumstances, the order of characters differs from this logical order when the text is displayed or printed. (...) For the most part, logical order corresponds to *phonetic order*. ([24], pp. 18-29)

Put simply, a sequence of *characters* in logical order is a *pre-text*, as mentioned above. What we need to remember here is that Unicode supposes the precedence of the spoken language over the written language. Although, such assumptions are common

[3] Papers on these projects may be included in this proceeding.

[4] Before version 4.0, the first principle was "Sixteen-bit character code." As known well, this principle did not make any sense from an earlier version.

to linguists and people, it's important as what specifies the limit of CCSM[5].

It should be noticed that what should be regarded as *logical* in the standard is not defined. However, only the rough relation with *phonetic order* is shown. Although, of course, it's very difficult to define what is logical, it would be defined if this principle were really put into practice.

Next for the fourth principle, the "semantic value" of a *character* mentioned above is defined in the quotation below:

> Characters have well defined the semantics. Characters property tables are provided for use in parsing, sorting, and other algorithms requiring semantic knowledge about the code points. The properties identified by the Unicode Standard include numeric, spacing, combination, and directionarity properties (...). Additional properties may be defined as needed from time to time. ([24], pp. 17-18)

Here, it's explained what constitutes an abstract *character*. It's of value to point out that semantics which attribute to a *character* must be "well defined." Thus the distinction between *characters* and *glyphs* means that Unicode regards glyphs as accidental.

If that's true, will including glyph information into the semantic values of *characters* solve the problems? —Probably not. The property relevant to context-dependent glyph behavior is already defined (see *Bidi Mirrored*, [24], p. 101). As discussed later, the true problem is not an informational quantity but distinction between essential and accidental.

Lastly, for the seventh principle, there is an explanation about the famous unification principle of Unicode, which has attracted a lot of criticism and created a lot of misunderstanding, until now:

> The Unicode Standard avoids duplicate encoding of characters by unifying them within scripts across languages; characters that are equivalent are given a single code. (...) Avoidance of duplicate encoding of characters is important to avoid visual ambiguity. ([24], pp. 19-20)

Although stated "visual ambiguity" here, the definition of *character* seems more ambiguous to me.

The term *script* in this principle is a concept abstracted from a language: "Script. A collection of symbols used to represent textual information in one or more writing systems" ([24], p. 1377). This is a variation of the concepts already seen.

To sum up these principles:

1. Unicode is for describing logical characters and texts.
2. Unicode regards the oral language as logical rather than visually rendered glyphs.

2.2 Aristotle's Essentialism and Unicode.

For our investigation, it would be helpful to compare these definitions of Unicode with the essentialism of Aristotle and his followers. Aristotle says:

> The formula (*logos*) of the essence of x is a formula in which x itself does not appear but which expresses (*legonti*) x. ([1], 1029b19-20[6])
>
> A definition is an account (*logos*) that signifies an essence. ([3], 102a3)

That is, Aristotle states that the essence (*to ti ēn einai*: the "what it was meant to be") of a thing is equal to the definition (*logos*) of it[7]. It should be noted that Aristotle's theory of essence was also established as a criticism to Socrates and Plato who considered that the essence of words dependent on contexts, such as beauty. Essence is not influenced by any contexts.

It seems clear that his way of thinking resembles the definition of *characters* in Unicode. Thus, in the

[5] What shown in this principle may be associated with I. J. Gelb's "theory of writing." Gelb expects that all writing systems would converge theoretically on something like IPA in the future: "What is needed now is one system of writing in which signs have identical or almost identical phonetic correspondences all over the world. That need is fulfilled in the IPA alphabet" ([10], p. 241).

[6] Page numbers of Aristotle's works are traditionally based on so-called "Bekker edition" (*Aristotelis opera, edidit Academia regia borussica*, edited by Immanuel Bekker. 5 vols. Berlin, G. Reimer, 1831-1870.)

[7] For further investigation beyond this paper, it should be memorized that an essence is in apposition with a necessity in modal logic. If A is the essence of B, B is necessarily A. I argued once that the feature of the Chaon model disscussed below differed from the definite description proposed by Bertrand Russel ([19].)

context of Unicode, a code point is definiendum and "well defined semantics" is definiens. A *character* is logical and precedes an actual character.

Moreover, this concept is closely linked with the sixth (*Logical* Order) of the ten design principles. In that principle, which we have discussed above, the precedence of the spoken language has been taken over the written language. Aristotle also stated the similar hierarchy between the spoken and written words:

> Spoken words (*ta en tē phonē*) are the symbols of mental experience (*pathēma tes psychēs*) and written words are the symbols of spoken words. ([2], 16a3)

Both Aristotle and Unicode consider that the spoken language is closer to the soul rather than the written language. According to Jacques Derrida who is a very famous philosopher of deconstruction, the logicalness and the proximity to the spoken language are closely related:

> Within this logos, the original and essential link to the *phonè* has never been broken. (...) the essence of the *phonè* would be immediately proximate to that which within "thought" as logos relates to "meaning," produces it, receives it, speaks it, "composes" it. ([6], p. 11)

The fundamental difference between the spoken and written language is that the former is dependent only on the context at the time of utterance, although the latter may be read apart from the intention of a writer in another context. As well as tautology being one of the foundations of logic, it's regarded as logical that the idea of a speaker is repeated by his speaking. When the visual glyph rendering is included within the intention of a writer, correct glyph rendering would be regarded as logical and properties for correct rendering would be contained in the "semantic value." Conversely, in the context of essentialism, what is not essential cannot be defined.

Thus, it's dependent on the arbitrariness of architects what is made into essence. This issue is surely not irrelevant to the fact that this standard has been developed relying not only on various local standards but also on authorized *great texts*[8], such as *Kangxi Dicionary* for CJK repertoires. The argument on the essence of the character or the language is easily connected with authoritarianism and nationalism (e.g. what's the essence of Japanese?) For example, according to Taichi Kawabata's critical study, the unification rule was changed for the convenience of Taiwanese national standard (and the inconvenience of Japanese standard) when CJK Unified Ideograph Extension B was defined ([13]).

[8]For more discussion about *great texts*, see [15].

3 What crosses borders?

3.1 Reason of the Essentialism.

Up to this point I have performed an overview of the philosophical problems of the Unicode Standard. This might be applicable to other character codes based on CCSM. Consider now the background for which such an abstract *character* was invented.

The ontological assumption for computing, which Dreyfus states in his criticism to AI, may serve to consider it:

> (...) since all information fed into digital computers must be in bits, (...) all relevant infomation about the world, everything essential to the production of intelligent behavior, must in principle be analyzable as a set of situation-free determinate elements. ([8], p. 156)

Dreyfus also states that such assumptions can be located on the same line of the history of European philosophy which starts with Plato, via Leibniz and results in Wittgenstein's *Tractatus Logico-Philosophicus* and there has been "the Merleau Pontyan, Wittgensteinian and Heideggerian critique of the traditional ontological assumption." ([8], pp. 211-212.) Although I think it's not strict, since he does not refer to the importance of Aristotle's criticism of Plato, Jacques Derrida's deconstruction of European philosophical traditions and so forth. His argument is seen to be of value for our discussion. Especially Leibnitz is important according to his close relation with Computer science.

In addition, it seems extremely inefficient or impossible for both computers and us to take all contexts into consideration for the advance for our computers (TRON code is realized by disregarding almost all contexts). Efficiency is also one of the prin-

ciples of Unicode ([24], p. 15.) When the performance of personal computers was poor or modeling only a phonogram like the alphabet into a computer, CCSM might be the best way to handle characters.

Although all these topics are attractive for us, the primary consideration should be the relationship between characters / texts and contexts. As I stated at the beginning of this paper, this issue has begun to receive more attention mainly because of today's international communication following the advent and explosive propagation of the Internet.

3.2 Surface or Essence.

As I stated above, there are two sides in a character: depending on the context and exceeding it. In other words, a character is polysemous.

Based on the essentialism of *character*, the relationship between characters and contexts is explained as follows: If the context around a character changes, its glyph, pronunciation(s,) meaning(s) and etc. may change. However the essence of the character, which does not change with context, exists in the background of its polysemy. What changes in a character is an accidental, in other words, a surface part. Surface polysemy is supported by and derived from the essence.

To borrow an argument on Saussure's general linguistics from Hiroki Azuma ([4], p. 34-36), here we find an aporia: The arbitrariness of binding a signifiant (signifier) and a signifié (signified) is dependent on the comparison of different languages. Actually, signifié can be known only by signifiant in each language. For example, a glyph 一 in Chinese signifies some semantics and a glyph 一 in Japanese also signifies some semantics. However there is no basis, which guarantees to know that both glyphs have the same semantics, since Chinese and Japanese are respectively independent systems. The issue on languages is applicable to that of contexts. The idea of the essence, which is behind different contexts, is based on the same aporia.

Alternatively, it could be argued that the essence does not move but character as only a substance does: When a character is written first, it has only one semantic. Then, while passing two or more contexts, the character acquired polysemy. In other words, it is reasonable to say that what crosses borders is not the essence but the surface of a character.

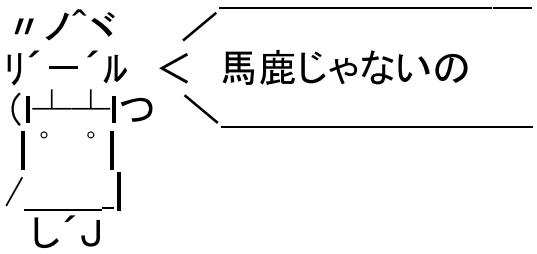

Figure 1: リル子 (Riruko).

The simplicity of a character is always earlier than the polysemy of a character.

Only the surface of characters may be needed in a certain situation. For example, Riruko (Fig. 1) is an ASCII art[9] describing a girl, by an anonymous artisan in a huge BBS named 2ch. Needless to say, when this figure was drawn, meanings and pronunciations of characters dropped out and only their shapes in the extremely limited font environment were expected. However, her name was derived from the half-width Katakana characters, *ri* リ and *ru* ル, used to express her sideburns. Thus, the phonetic / logical aspects of these characters were found after drawing.

My thesis owes a lot to Jacques Derrida and Hiroki Azuma. Derrida criticizes "*logocentrism*: the metaphysics of phonetic writing (for example, of the alphabet) which was fundamentally —for enigmatic yet essential reasons that are inaccessible to a simple historical relativism— nothing but the most original and powerful ethnocentrism in the process of imposing itself upon the world, controlling in one and the same *order*" ([6], p. 3). As we have seen, logo-centrism, phono-centrism, ethnocentrism and so on are also found in Unicode. Derrida's methodology serves to reveal the essential misapprehension of CCSM.

In the case of arguing about the polysemy of écriture, he distinguishes between *polysémie* and *dissémination* clearly ([5]). Dissémination, in other words mutiple belonging to contexts, is quite different from *polysémie* based on the context-

[9] Although originally the word "ASCII art" means face marks written in ASCII characters such as `:-)`, illustrating using 2-byte characters is also called "ASCII art" in Japan and is being developed dramatically especially in a Japanese anonymity BBS named "2ch (2 ちゃんねる)" (`http://www.2ch.net/`).

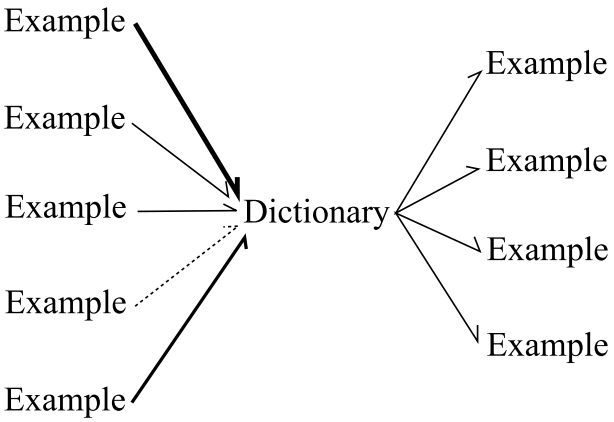

Figure 2: Relationship between examples and a dictionary.

independency of *characters*.

3.3 Metaphysics of Dictionary.

Let us begin to re-examine this issue using the relationship between a dictionary and examples (Fig. 2).

If we're asked which precedes, examples or a dictionary? Needless to say, we will answer that examples do. As for the order of our dictionaries, however, it seems that examples follow the definitions in a dictionary, and most of us accept it. Nevertheless, examples in different contexts are collected initially and then a dictionary is created.

If the dictionary is completed, we however read, write and learn characters depending on the dictionary. It's probable to think that the situations of our usual writing activity are within the right half of Fig. 2. That is, we usually depend on a dictionary unconsciously when reading or writing.

However we sometimes encounter situations like the left half of the figure. Unconscious or Intentional misuses are frequently performed especially in creative works. For example, new Chinese characters are created for names of newborn children in China even now. Moreover, a philologist (like me) who reads old manuscripts everyday often encounters characters which precede any dictionaries.

It's safe to say that the concept of *character* in Unicode also has the same structure as the hierarchy of dictionaries. Accordingly, it's highly probable that this is one of the reasons why we cannot use characters out of a character set. The context of writing in Unicode is strongly specified to the intention of the designer, although the regulation is hidden and authorized by the logicalness = dictionarity of Unicode. That is, writing based on CCSM is equal to the disability of writing without dictionaries.

Conversely, it should be required for a new character model to have the capability, which a character moves to a new context, in other words, the capability to produce, memorize and express it.

4 Introduction to the Chaon Model.

4.1 CHISE Project.

Thus far, we have outlined some philosophical background and problems of Unicode and now I would like to introduce the Chaon model of the CHISE Project[10].

The CHISE (CHaracter Information Service Environment; 知世) Project[11] attempts to develop a new environment for character / text processing, which is independent of CCSM, by collecting and organizing knowledge of characters. Dr. Tomohiko Morioka, who proposed the UTF-2000 model (now called Chaon model,) and his friends started this project. Now the project is constituted by several sub-projects listed below:

- Implementations of the Chaon model
  - libchise (common APIs to access Character Knowledge Database)
  - XEmacs/CHISE
  - Ruby/CHISE
  - Perl/CHISE[12]
  - Ω/CHISE
  - KAGE (a dynamic Kanji glyph generation system)
- Character Knowledge Database

[10] `http://www.kanji.zinbun.kyoto-u.ac.jp/projects/chise/`, `http://cvs.m17n.org/chise/`, `http://mousai.as.wakwak.ne.jp/projects/chise/`. See also [16]

[11] The project was called "UTF-2000 Project" before.

[12] I am a developer of Perl/CHISE. See [20].

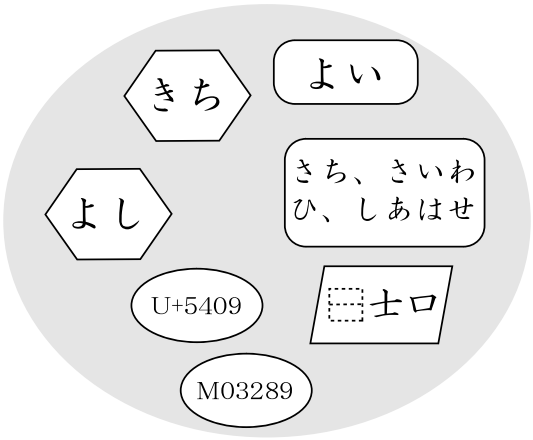

Figure 3: A character object based on the Chaon model

- Development of Character Knowledge Database(s)[13]
- Organization of Character Knowledge Database(s) based on TopicMaps[14]
- Mathematical analysis and visualization of character knowledge

Development and discussion in CHISE project are open to the Internet by using mailing lists and CVS. Most of these results are being published as free software under the terms of GNU General Public License.

4.2 Chaon Model.

The Chaon model is a character model of the CHISE project, which aims at an essential breakaway from CCSM. Although all the problems described so far could not be solved with this model, I believe that it would bring us a quite better computing environment.

In the Chaon model, a character is composed by a set of features[15], in other words, by a set of examples. Fig. 3 illustrates how the features of a character (吉 for example) are organized[16]. The view of regarding a character as a set of elements is not new. In the tradition of studies on Chinese characters, *liu shu* 六書 (six types of characters) is a popular method to interpret a character as a composition. Tatsuo Nishida, a famous linguist in Japan who attaches importance to the character, claims that *liu shu* may be applicable to analysis of all characters ([14], p. 154), although he gives precedence to the spoken language over the written language.

It should be noted that a "feature" does not mean an "attribute" of a character but something like a footmark which a character has traced because of two reasons: First, there are no central existences like the code point in CCSM. Second, users can change the contents of a set.

In the Chaon model, DBMS(s) stores features of characters. Although the local system database only exists now (using Berkeley DB as a backend now,) Dr. Morioka announced a future plan of the CHISE project enabling to use two or more DBMS, such as private databases in local hard disks or public databases on the web, through libchise.

If we could build such an environment, where we could write a character which does not exist in any systems, we could write a character, in quite a new way which is resently not in use. Registering a set of features into a database or changing the contents of a character could instantly use this character. By contraries, when all features are removed from a character object, the character object will disappear (In CCSM, even if all attributes are lost, a code point remains.)

Of course, this model needs a rich computer environment. It can process enough, however, for our present personal computers and network environment.

A computer must prepare data and algorithma in advance of processing. In other words, modeling for character processing is not handling the actual characters, contexts and so on but pre-characters and pre-contexts before writing. Even though, issues such as glyph representation and/or context-dependent processing are discussed for modeling. In this point, there is no difference between CCSM and

[13] Taichi Kawabata and I manage Kanji Database Project (`http://kanji-database.sourceforge.net/`), which is also developing a fundamental Kanji database for appropriate text processing.

[14] `http://www.topicmaps.org/`

[15] Prof. Naoki Yamazaki (Osaka University of Foreign Studies), in a personal discussion, pointed out that the Chaon model is similar to the phonological theory of distinctive features' set proposed by R. Jakobson ([12].) Although it is, strictly speaking, different from the Chaon model, their comparison may be helpful to us.

[16] Although Fig. 3 includes features on pronunciations and meanings, they have not been stored in the officially released database.

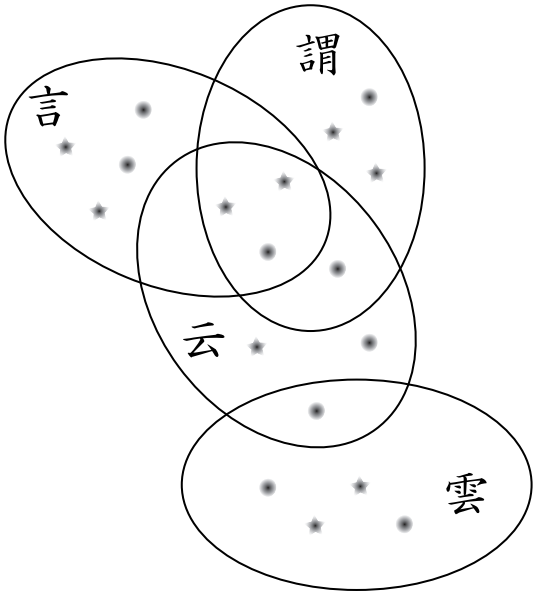

Figure 4: Set operation between characters.

the Chaon model. However, the Chaon model is designed to be able to add features at any time and it's just a deterministically different point from CCSM. As for Fig. 2, the left half is also modeled in the Chaon model.

In addition, although this set of features is called a character "object," this term *object* does not mean that of the object-oriented model[17].

4.3 Comparison of Characters as Set Operation.

In the Chaon model, a character is expressed by a set of features, comparison between characters are regarded as a kind of set operation[18]. A test implementation of the set operation can be seen in Perl/CHISE ([19][16]).

In Fig. 4, areas enclosed by a circle mean character objects and small dots mean features. Three upper Chinese characters (言, 謂, 云) have the same meaning such as speaking or to say, but their pronunciations are different. Although the bottom Chinese character (雲) means clouds and is quite different from three upper ones, two lower Chinese characters have the same pronunciation. Thus, in the People's Republic of China, the second Chinese character from the bottom is used as the simplified character of the bottom one. Thus, if a context changes, a comparison result will also change.

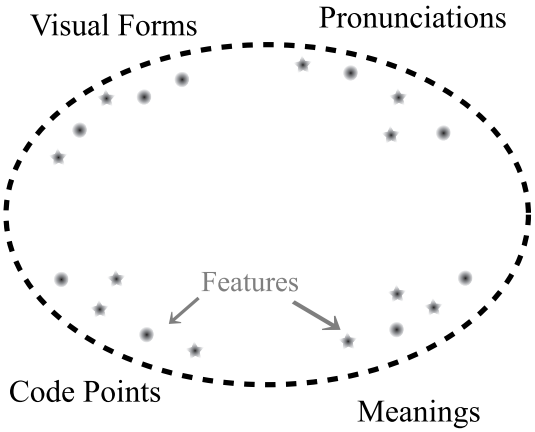

Figure 5: set of surfaces.

Moreover, the union of two or more character objects means the unification of characters in the context of the Chaon model.

Needless to say, since an actual relationship of features might not be a simple set like Fig. 3, but should be described as a kind of hierarchy or network. Comparison between character objects should also be more complex. The Chaon model however, has only a simple structure yet. Presently, solutions to describe and handle the semantic structure of characters by TopicMaps are being studied.

4.4 Set of Surfaces.

When written on paper, glyphs, shapes and stain of ink or so are the surface of characters. In the context of the Chaon model, however, each feature may be regarded as surface[19], since the physical nature of writing in computers differs from that on paper.

[17] Prior to, using the concept of *khôra* of J. Derrida ([7]), I used the expression "a transparent vessel" in order to explain a nature of the character object of the Chaon model. I then received an indication from Dr. Morioka in a personal communication that a character object of the Chaon model differs from that of the object-oriented model. We can find that there are several people who point out that the object-oriented model is similar to essentialism. For example, see [9] and [11].

[18] Masaaki Nomura proposes a similar method of comparison of the Chinese character by traditional three elements (shapes, pronunciations and meanings.) ([22], pp. 25-26.)

[19] I applied the Chaon model to the N-gram analysis by implementing a sample processing based on the phoneme features of Chinese characters ([21]).

Even now a sequence of code points is rendered not only visually but also vocally through speech synthesizer[20].

Thus it is decided by the context which feature will become a surface. If a certain feature becomes a surface, other features will hide in the depths and will function like the essence which produces polysemy (Fig. 5.)

It's reasonable to think that a code value also belongs to a surface in a circumstance of computing such as Unix redirection. This means that the Chaon model is not contradictory. In other words, it's upper-compatible to conventional CCSM. The plain text made only of code points may be interpreted by a Chaon-based system as a text written in the context of CCSM. Therefore CCSM is one of the contexts where a character of the Chaon model belongs.

5 Conclusion.

So far I have presented the problems based on the essentialism of CCSM and the philosophical posibility of the Chaon model that would release the fundamental handling of the character from the architects of coded character sets and the creators of dictionaries. From this point we might go on to investigate and develop a new writing environment.

In concluding, I should emphasize that the philosophical analysis on the character is valid for considering character models. From this we can derive the argument that it will become a new subject of Humanities in which computer use is remarkably incresing in recent years. The scholars of Humanities have so far criticized nationalism, orientalism and so forth. In the same way they have to critically examine the models used in computers, although the use of information technology is going to be accepted uncritically and obsessively.

Moreover, I would like to point out the importance of implementation, since the monopoly of CCSM is not healthy. Diversity induces development. It will be healthy and very much appreciated to criticize this paper based on the points of view of Humanities, Computer science, hackers and so on.

[20] Taken in this light, we should focus on the movement of the web accessibility.

Bibliography

[1] Aristotle. *Metaphysics.*

[2] Aristotle. *On Interpretation.*

[3] Aristotle. *Topics.*

[4] Azuma, Hiroki. 存在論的、郵便的 —ジャック・デリダについて [*Ontological, Postal: On Jacques Derrida*]. Tokyo: Shinchosha. 1998.

[5] Derrida, Jacques. "Signature Événement Contexte." *Marges.* Minuit. 1972.

[6] Derrida, Jacques. *Of Grammatology*, translated by Gayatri Chakravorty Spivak. Johns Hopkins University Press. 1974.

[7] Derrida, Jacques. "Khôra." *On the Name*, translated by David Wood and John P. Leavey. Stanford University Press. 1995.

[8] Dreyfus, Hubert L. *What Computers Can't Do: The Limits of Artificial Intelligence* (Revised Edition). New York: Harper & Row. 1979.

[9] Fleishman, Stiv. "On Aristotelian Syllogistic (SylL)." [Online] Available: `http://home.gwu.edu/~stiv/p45f02n6p1.html` (20, Jun., 2000).

[10] Gelb, Ignace J. *A Study of Writing* (Revised Edition.) University of Chicago Press. 1963.

[11] Hayakawa, Tetsuro. "アリストテレス的オブジェクト指向入門 [Introduction to Object-oriented Model with Aristotle]." [Online] Available: `http://itpro.nikkeibp.co.jp/NIP/nip04_bn.jsp?BN=Y&OFFSET=1` (27, Nov., 2002-24, Jul., 2003.)

[12] Jakobson, Roman, G. Fant and M. Halle. "Preliminaries to speech analysis." Technical Report 13, Acoustic Laboratory, MIT. 1952.

[13] Kawabata, Taichi. "新 ISO/IEC 10646 と Unicode の漢字を検証する [A Study on the Chinese Characters of ISO/IEC 10646-2 and Unicode 3.1]." *Journal of the Japan Association for East Asian Text processing*, vol. 2. 2001.

[14] Kohno, Rokurou and Nishida, Tatsuo. 文字贔屓—文字のエッセンスをめぐる3つの対話 [*Character Favor: Three Dialogues on the Essence of Characters*]. Tokyo: Sanseido. 1995.

[15] LaCapra, Dominick. *Rethinking Intellectual History: Texts, Contexts, Languages*. Cornell University Press, 1983.

[16] Morioka, Tomohiko, et al. "CHISE Project" (in Japanese). *Journal of the Japan Association for East Asian Text processing*, vol. 4. 2003.

[17] Moro, Shigeki. "タグ付き言語と文字コード [Language Attributes in XML and Character Code Sets]." Kobayashi, Tatsuo et al. インターネット時代の文字コード. Tokyo: Kyoritsu Shuppan. 2001.

[18] Moro, Shigeki "GB18030 とは何か 大陸の戦略 [What is GB18030? —Strategy of China]." *Journal of the Japan Association for East Asian Text processing*, vol. 2. 2001.

[19] Moro, Shigeki. "Chaon モデルの過去・現在・未来 [Past, Present And Future Of The Chaon Model]," presented At Chise Symposium 2003 (15, Mar., 2003.)

[20] Moro, Shigeki. "Perl/CHISE による正規表現の拡張の試み—文字素性による後方参照の実装実験と課題— [A Trial of Extension of Regular Expression with Perl/CHISE: Implementation of Backreference Function by Character Features and its Problems]." Linux Conference 抄録集: 第1巻 (2003年), ISSN 1348-7876, `http://lc.linux.or.jp/paper/lc2003/`

[21] Moro, Shigeki. "N グラムと文字データベースによる漢字仏教文献の分析 [Analysis of Chinese Buddhist Texts with N-gram Model and Character Knowledge Database]." *IPSJ SIG Notes*, No. CH-61-3, Tokyo: IPSJ. 2004.

[22] Nomura, Masaaki. "同字と別字のあいだ [Between Same Characters and Different Characters]." *Nihongogaku*, 3-3. Tokyo: Meijishoin. 1984.

[23] "TRON コード概要 [Introduction to TRON Code]." [Online] Available: `http://www2.tron.org/troncode.html` (No Date.)

[24] Unicode Consortium. *The Unicode Standard, Version 4.0*. Boston: Addison-Wesley, 2003.